ACKNOWLEDGMENTS

This research was made possible by many people that provided support, encouragement, and advice. I am very grateful to this team.

All the help and support that my advisor, Dr. Alireza Khotanzad, provided, deserves a very special recognition. His advice, guidance, and assistance made this work possible. His teachings, insight, coaching, and experience were invaluable resources during this research. A special recognition is also well deserved by Dr. Jesse Bennett. I am extremely grateful for the many hours he put forth in discussion of ideas. I would also like to thank the rest of my supervisory committee, Dr. Carlos Davila, Dr. Someshawar Gupta, Dr. William Schucany, and Dr. Mandyam Srinath for reading and advising on the manuscript of this dissertation.

My greatest thanks and appreciation go to my wife Ana, my son Orlando, and my daughter Anelis. Their love, understanding, patience, and support were crucial during this work. I thank them and admire them, because they beforehand forgave me for all the weekends that I stole from them to achieve this work. For this, I will be eternally grateful.

Hernandez, Sr., Orlando J. B.S.E.E., University of South Florida, 1991
M.S.E.E., University of South Florida, 1993

Color Image Retrieval Using Multispectral
Random Field Texture Models and
Color Content Features

Advisor: Professor Alireza Khotanzad

Doctor of Philosophy conferred May 18, 2002

Dissertation completed April 26, 2002

This work describes a color texture-based image retrieval system for query of an image database to find similar images to a target image presented during the query action. The entire retrieval process is automatic, and the only requirement is input of the query image. The color texture information is obtained via modeling with the Multispectral Simultaneous Auto Regressive (MSAR) random field model. The general color content characterized by ratios of sample color means is also used. The retrieval process involves segmenting the image into regions of uniform color texture using an unsupervised histogram clustering approach that utilizes the combination of MSAR and color features. The color texture content, location, area, and shape of the segmented regions are used to develop similarity measures describing the closeness of a query image to database images. These attributes are derived from the maximum fitting square and best fitting ellipse to each of the segmented regions. The proposed similarity measures combines all these attributes to rank the closeness of the images. The performance of the system is

tested on three databases containing synthetic mosaics of natural textures and natural scenes, respectively.

TABLE OF CONTENTS

ACKNOWLEDGMENTS iv

LIST OF TABLES x

LIST OF FIGURES xii

Chapter

1. INTRODUCTION 1

    1.1 Research Objectives 2

    1.2 Organization of the Dissertation 4

2. RELATED STUDIES 6

3. TEST DATABASES 12

    3.1 Natural Texture Mosaics Database I 12

    3.2 Natural Texture Mosaics Database II 15

    3.3 Natural Scenes Database 17

    3.4 Conclusions 19

4. TEXTURE CHARACTERIZATION BY COLOR TEXTURE AND COLOR CONTENT FEATURES 20

    4.1 Color Texture Characterization with Multispectral Simultaneous Autoregressive Model 20

        4.1.1 Least Squares Estimation of MSAR Model Parameters 22

        4.1.2 MSAR Based Texture Synthesis 24

    4.2 Color Content Characterization 25

    4.3 Conclusions 27

5. UNSUPERVISED SEGMENTATION WITH A HISTOGRAM-BASED CLUSTERING ALGORITHM 28

5.1 Feature Extraction with a Sliding Window 28

5.2 Clustering Algorithm 30

5.2.1 Automatic Selection of Quantization Level, Cluster Validation, and Automatic Selection of Window Size 32

5.3 Experimental Results 34

5.4 Conclusions 39

6. ATTRIBUTES USED IN SIMILARITY METRICS 40

6.1 Image Separation into Regions 40

6.2 Metrics Description 41

6.2.1 Texture-Based Metrics 41

6.2.2 Shape-Based Metrics 41

6.3 Conclusions 46

7. THE RETRIEVAL PROCESS 47

7.1 Search Space Reduction 47

7.2 Region Association 49

7.3 Similarity Computation 50

7.4 Retrieval 52

7.4.1 Overall System Description and Retrieval Algorithm 53

7.5 Conclusions 55

8. EXPERIMENTAL RESULTS 56

8.1 Experimental Results for the Search Space Reduction 56

8.1.1 Experiments using a Natural Textures Database 56

8.1.2 Experiments using a Natural Scenes Database 58

8.2 Retrieval Results 59

8.2.1 Experiments using Natural Texture Mosaics Database I 59

8.2.2 Experiments using Natural Texture Mosaics Database II 61

8.2.3 Experiments using the Natural Scenes Database.................... 72

8.3 Performance Evaluation 81

8.4 Conclusions 84

9. SUMMARY AND CONCLUSIONS 85

9.1 Directions for Future Research 87

APPENDIX

A. NATURAL TEXTURE MOSAICS DATABASE II 89

B. NATURAL SCENES DATABASE 95

REFERENCES 106

LIST OF TABLES

Table		Page
1.	Components of the C^3T Feature Set.	27
2.	Sample of Characterization Parameters for Figure 17.	45
3.	S Values for Retrieval Results Shown in Figure 22.	60
4.	S Values for Retrieval Results Shown in Figure 23.	62
5.	S Values for Retrieval Results Shown in Figure 24.	63
6.	S Values for Retrieval Results Shown in Figure 25.	64
7.	S Values for Retrieval Results Shown in Figure 26.	65
8.	S Values for Retrieval Results Shown in Figure 27.	66
9.	S Values for Retrieval Results Shown in Figure 28.	67
10.	S Values for Retrieval Results Shown in Figure 29.	68
11.	S Values for Retrieval Results Shown in Figure 30.	69
12.	S Values for Retrieval Results Shown in Figure 31.	70
13.	S Values for Retrieval Results Shown in Figure 32.	71
14.	S Values for Retrieval Results Shown in Figure 33.	73
15.	S Values for Retrieval Results Shown in Figure 34.	74
16.	S Values for Retrieval Results Shown in Figure 35.	75
17.	S Values for Retrieval Results Shown in Figure 36.	76

18. S Values for Retrieval Results Shown in Figure 37. 77

19. S Values for Retrieval Results Shown in Figure 38. 78

20. S Values for Retrieval Results Shown in Figure 39. 79

21. S Values for Retrieval Results Shown in Figure 40. 80

22. Performance Analysis Results. 82

LIST OF FIGURES

Figure Page

1. Images in the Natural Textures Mosaics Database I. 13

2. Samples of Images in the Natural Textures Mosaics Database II. 16

3. Samples of Images in the Natural Scenes Database. 18

4. Neighbor Set Used with the MSAR Model. 23

5. Neighbor Set Used for Image Synthesis Results in Figure 6. 24

6. MSAR Model Image Synthesis Results. Top Row: Original Color Texture Images. Bottom Row: Synthesized Images Using MSAR Parameters Estimated from Original Images. 25

7. Feature Vector Extraction with a Sliding Window. 29

8. Illustration of the Peak Climbing Approach for a Two-Dimensional Feature Space Example. 31

9. Illustration of Color Texture Image Segmented into Regions. 34

10. Segmentation Results for Four Images from the "Natural Textures Mosaic Database II". 1st row: Original Image. 2nd row: Segmentation Results. 3rd row: Texture Boundaries Corresponding to Segmentation Results. 4th row: Segmentation Using the JSEG Method. 36

11. Segmentation Results for Four Images from the "Natural Scenes Database". 1st row: Original Image. 2nd row: Segmentation Results. 3rd row: Texture Boundaries Corresponding to Segmentation Results. 4th row: Segmentation Using the JSEG Method. 37

12. Segmentation Results for Another Set of Four Images from the "Natural

Scenes Database". 1st row: Original Image. 2nd row: Segmentation Results. 3rd row: Texture Boundaries Corresponding to Segmentation Results. 4th row: Segmentation Using the JSEG Method. 38

13. Segmentation Result for a Natural Scene. 39

14. Example of an Image with Two Textures but Three Regions. 40

15. Examples of Texture and Shape Attribute Computation for Texture Mosaic Images. 43

16. Examples of Texture and Shape Attribute Computation for Natural Scenes... 44

17. Image for Characterization Parameters in Table 2. 45

18. An Example of the Region Association Process. 49

19. Overall System Block Diagram. 54

20. Search Space Reduction Results for a Natural Texture Mosaics Database. 57

21. Search Space Reduction Results for a Natural Scenes Database. 58

22. Retrieval Results for Natural Texture Mosaics Database I. The Image at the Top with Blue Border Is the Query Image. All Other Images Are Ranked in the Order of Similarity with the Query Image from Left to Right, Top to Bottom. 60

23. First Example of Retrieval Results for Query of the Natural Texture Mosaics Database II. The Image at the Top with Blue Border Is the Query Image. All Other Images Are Ranked in the Order of Similarity with the Query Image from Left to Right, Top to Bottom. 62

24. Second Example of Retrieval Results for Query of the Natural Texture Mosaics Database II. 63

25. Third Example of Retrieval Results for Query of the Natural Texture Mosaics Database II. 64

26. Fourth Example of Retrieval Results for Query of the Natural Texture Mosaics Database II. 65

27. Fifth Example of Retrieval Results for Query of the Natural Texture

Mosaics Database II. 66

28. Sixth Example of Retrieval Results for Query of the Natural Texture Mosaics Database II. 67

29. Seventh Example of Retrieval Results for Query of the Natural Texture Mosaics Database II. 68

30. Eighth Example of Retrieval Results for Query of the Natural Texture Mosaics Database II. 69

31. Ninth Example of Retrieval Results for Query of the Natural Texture Mosaics Database II. 70

32. Tenth Example of Retrieval Results for Query of the Natural Texture Mosaics Database II. 71

33. First Example of Retrieval Results for Query of the Natural Scenes Database. The Image at the Top Left with Blue Border Is the Query Image. The Other Images from Left to Right, Top to Bottom Are Retrieved Images with Decreasing Similarity. 73

34. Second Example of Retrieval Results for Query of the Natural Scenes Database. 74

35. Third Example of Retrieval Results for Query of the Natural Scenes Database. 75

36. Fourth Example of Retrieval Results for Query of the Natural Scenes Database. 76

37. Fifth Example of Retrieval Results for Query of the Natural Scenes Database. 77

38. Sixth Example of Retrieval Results for Query of the Natural Scenes Database. 78

39. Seventh Example of Retrieval Results for Query of the Natural Scenes Database. 79

40. Eighth Example of Retrieval Results for Query of the Natural Scenes Database. 80

To My Wife, Ana, My Children, Orlando and Anelis, My Parents, Maria and Ambrosio,

My Brother Frank, and My Sister Jackie

Chapter 1

INTRODUCTION

An ever-increasing usage of digital images and the wide acceptance of very large volume image databases naturally give rise to the difficult problem of organizing them for the best and rapid access to their information contents. And the ability of computers to rapidly and successfully retrieve information from image databases based on the objects contained in the images has a direct impact on the progress of digital library technology [1]. With the growth of multimedia computing and the spread of the Internet, more and more people have access to large databases and would have applications for such retrieval systems [2], [3], [4]. Retrieval of image data based on pictorial content queries is an interesting and challenging problem actively worked on by the research community [5], [6].

This work describes a color texture-based retrieval system that finds images that are similar to a query image in a large database. This query image is an input to the system, and the search and similarities computations are with reference to this image. In fact the only two inputs to the system are the query image, and the number of most similar images that the search should retrieve. Everything else in the retrieval process is automatic and unsupervised. The similarity criterion is based on the notion of containing same/similar color texture regions. The approach involves characterizing color texture using features derived from a class of multispectral random field models and color space. The color texture information is obtained via modeling with a class of random field models called the Multispectral Simultaneous Auto Regressive (MSAR) random field model [7], and the general color content is characterized by ratios of sample color means. The retrieval process involves segmenting the image into regions of uniform color texture using an unsupervised histogram clustering approach that utilizes the combination of MSAR and color features. After segmentation, the texture and color features are used in an unsupervised histogram clustering-based algorithm to find regions of similar texture to the query image regions in database images. This is how the retrieval process also involves identifying those images in the database that contain similar kinds of texture found in the query image. The system then considers the similarities in the size and spatial arrangement of the texture regions to carry out the final retrieval process. The color texture content, location, area, and shape of the segmented regions are used to develop similarity measures describing the closeness of a query image to database images. The proposed similarity measure combines all these attributes to rank the closeness of the images. The performance of the system is tested on three databases containing synthetic mosaics of natural textures and natural scenes respectively.

1.1 Research Objectives

Texture information has been used in the past for browsing and retrieval of imagery, however the previously proposed approaches have either considered only gray level textures or pixels based color content and not color texture [8]. The utilized segmentation algorithm in some approaches has also not been completely unsupervised [9].

One of the main algorithmic engines used in this work is a multispectral random field texture model. Bennett and Khotanzad [7] considered this multispectral extension to the traditional gray level simultaneous autoregressive (SAR) and Markov random field (MRF) models. These models are well suited for analysis and modeling of color image textures. In this work, these random field models, along with a set of color content features are used to characterize color texture regions.

The goal of this research is to overcome some of the current issues found in browsing and retrieval of large color image databases. The focus of this work is on development of new retrieval algorithms, as well as implementation and validation of the proposed algorithms. Experimental results are included with two types of databases: synthetic mosaics composed of natural textures, and natural scenes. These results demonstrate the effectiveness of the computational procedures and algorithm developed in this work. Specific goals of this research are:

- To extend and generalize the existing body of work on image segmentation, and develop an unsupervised image segmentation method that leverages both color and texture information.
- To develop a comprehensive system and a practical method that can be used to retrieve likely matches to a query image from a large color image database.
- To demonstrate the utility of the new developments with experimental results.

The contributions of this research are as follows:

- Texture characterization by a mix of multispectral random field based features and color content features.
- Development of a completely unsupervised color-texture-based segmentation algorithm and demonstration of its effectiveness.
- Development of a retrieval system based on novel similarity and ranking criteria that considers the type, size, location, and shape of various color-texture regions in the query and database images.
- Demonstration of the effectiveness of the proposed approach on two groups of databases containing 1) synthetic mosaics of natural textures and 2) natural scenes.

1.2 Organization of the Dissertation

The organization of this dissertation is as follows. First, Chapter 1 gives an introduction of the work that was carried out, including an overview of the limitations of current approaches, and the high level objectives of the research. Chapter 2 covers a survey of relevant research that is related to this research. In Chapter 3, the different databases that were used throughout this work are presented and explained. Chapter 4 describes the development of the C^3T (Color Content Color Texture) features that are used for image segmentation in Chapter 5. In Chapter 5, the utility of a class of random field models (the MSAR model) and color content to image segmentation has been addressed. In this chapter, a new method for image segmentation is investigated, and experimental results are shown to support the color texture image segmentation algorithm developed. Then in Chapter 6, the attributes for the characterization of the images, and used in the similarity metrics, have been addressed, and detailed results are shown to further illustrate these attributes. In Chapter 7, the complete color texture image retrieval process based on a contextual example and similarity metrics is described. This includes the calculation of the similarity metrics between previously segmented color textured images. Examples are shown in this chapter to further clarify the algorithm details to the prospective readers. In Chapter 8, experimental results are shown to support the retrieval algorithm that was developed based on the devised image metrics for image similarity calculations. Additionally, Chapter 8 presents performance evaluations that were conducted for the overall retrieval system. Finally, Chapter 9 offers conclusions, and some reflections regarding directions for future research that could be based on this work.

Chapter 2

RELATED STUDIES

There are two main aspects to this work. One is segmentation of color texture images, and the other is the retrieval of like images from a database based on a query image. Accordingly, previous studies related to these two topics are reviewed in this chapter.

Image segmentation is a popular field of research that has been covered extensively in the literature, for many years, and up to very recent publications. For example, the texture segmentation algorithm in [10] considers features extracted with a 2-D moving average (MA) approach. The 2-D MA model represents a texture as an output of a 2-D finite impulse response (FIR) filter with simple input process. The 2-D MA model is used for modeling both isotropic and anisotropic textures. The maximum likelihood (ML) estimator of the 2-D MA model is used as texture features. Supervised and unsupervised texture segmentation are considered. The texture features extracted by the 2-D MA modeling approach from sliding windows are classified with a neural network for supervised segmentation, and are clustered by a fuzzy clustering algorithm for unsupervised texture segmentation.

Mirmehdi and Petrou in [11] present an approach to perceptual segmentation of color image textures. Initial segmentation is achieved by applying a clustering algorithm to the image at the coarsest level of smoothing. The image pixels representing the core clusters are used to form 3D color histograms that are then used for probabilistic assignment of all other pixels to the core clusters to form larger clusters and categorize the rest of the image. The process of setting up color histograms and probabilistic reassignment of the pixels to the clusters is then propagated through finer levels of smoothing until a full segmentation is achieved at the highest level of resolution.

Much work can be found in the literature regarding matching and retrieval of color patterns from a database. For instance, Mojsilović *et al.* [12], [13], propose a perceptually based system for pattern retrieval and matching. The central idea is to model similarity judgment along perceptual dimensions. They detected basic visual categories that people use in judgment of similarity, and designed a computational model that accepts patterns as input, and depending on the query, produces a set of choices that follow human behavior in pattern matching. To understand how humans perceive color patterns, they performed a subjective experiment. The experiment yielded five perceptual criteria used in comparison between color patterns (vocabulary), as well as a set of rules governing the use of these criteria in similarity judgment (grammar). Following the processing typical for human vision, they designed a system to: a) extract perceptual features from the vocabulary and b) perform the comparison between the patterns according to the grammar rules. They proposed new color and texture features, as well as new distance functions that correlate with human performance. The performance of the system was illustrated with numerous examples from image databases from different application domains.

Saber and Tekalp study in [14] presents algorithms for automatic image annotation and retrieval based on color, shape, texture, and any combination of two or more of these features. Pixel- or region (object)-based color; region-based shape; and block- or region-based texture features were considered. Automatic region selection was accomplished by integrating color and spatial edge features. Color, shape, and texture indexing was knowledge-based (using appropriate training sets) or by example. The multi-feature integration algorithms were designed to: i) offer the user a wide range of options and flexibility in order to enhance the outcome of the search and retrieval operations, and ii) provide a compromise between accuracy and computational complexity, and vice versa. The authors demonstrated the performance of the proposed algorithms on a variety of images. It should be noted that this method uses supervised pattern recognition scheme whereas the approach described in this work is based on completely unsupervised approaches.

Fuh *et al.* [15] propose a model for content-based image retrieval using the idea of combining color segmentation with relationship trees and a corresponding tree-matching method. They retain the hierarchical relationship of the regions in an image during segmentation. In retrieval, they compare not only region features but also region relationships.

Retrieval of multimedia data from a database, based on a seed or example for the search/query requires the use of similarity metrics and similarity assessment. Santini and Jain look at this issue in [16]. They propose a definition of similarity as an operation, and develop a similarity measure, based on fuzzy logic, that exhibits several features that match experimental findings in humans. The model was dubbed *Fuzzy Features Contrast*

(FFC) and is an extension of the Feature Contrast model due to Tversky [17]. It is also shown how the FFC model can be used to model similarity assessment from fuzzy judgment of properties, and the use of fuzzy measures to deal with dependencies among the properties is addressed.

Adjeroh and Lee in [18] presented some techniques for improving the retrieval efficiency in image-based information systems, with performance guarantees on the reliability of results. Using the statistical theory of occupancy, they developed a model for the formal selection of the minimal subset of image features to be involved in histogram-based similarity evaluation. This guaranteed that decisions based on the minimum proportion are always the same as (or close to) the one that would have been reached by considering all the features. Results on real and simulated data showed the performance of the model on speedup, robustness, scalability, and performance guarantees.

Li *et al.* [19] also presented a novel similarity measure for region-based image similarity comparison called IRM (Integrated Region Matching). The images are represented by a set of regions, roughly corresponding to objects, which are characterized by features reflecting color, texture, shape, and location properties. The IRM measure for evaluating overall similarity between images incorporates properties of all the regions in the images by a region-matching scheme. Compared with retrieval based on individual regions, the overall similarity approach reduces the influence of inaccurate segmentation, helps to clarify the semantics of a particular region, and enables simple querying interface for region-based image retrieval systems.

Li, Chan, and Wang [20] presented a new method used in image classification and retrieval. This method was called the *nearest feature line (NFL)* method. Its performance was evaluated and compared with other methods by extensive experiments. The NFL method was demonstrated to make efficient use of knowledge about multiple prototypes of a class to represent that class.

Smith and Chang [9] have also presented a system for image retrieval using regions and their spatial and feature attributes. This method; however, does not make use of the texture information in the images, which sometimes results in image over-segmentation. Additionally, it is not a totally unsupervised query system, and the user has to input some a priori determined parameters for the search. The queries are not driven by the input of the query image to the system and the user has to enter a description for the image on line. The database has to be characterized for the possible types of queries that can be submitted as well; this is called the "Ground Truth Database". The approach described in this work does not have any of these limitations.

Research has also taken place on the performance of combined texture and color features for image retrieval, rather than just color or texture features alone. Yang and Chan [21] proposed the use of LHS color coordinate system, which is based on the Maxwell triangle of the RGB space. The luminance component is used for textural analysis. The hue and saturation components are used to retrieve similar images. Experiments were conducted, and showed that the image retrieval using the HS space has better precision and recall than the image retrieval using the RGB space. Moreover, it was also shown that retrieval through the chromatic features has better performance than retrieval through textural features only.

The International Business Machines (IBM) corporation has implemented a commercial system called QBIC™ (Query by Image Content) [22]. This system lets the user make queries of large image databases based on visual image content – properties such as color percentages, color layout, and textures occurring in the images. Such queries use the visual properties of images, so that the user can match colors, textures and their positions without describing them in words. This system; however, does not allow the user to query the database by providing an image example as a query for image matching and retrieval as it does the approach presented in this work.

None of the above approaches or any other color texture image segmentation work known to the author have taken into consideration and combined into a single body of work or framework the segmentation of the images using color texture modeling and color features, and multi-dimensional histogram feature separation. Additionally, none of the works known to the author augments this image segmentation approach with geometric features of the regions on the segmented images to achieve image matching and retrieval from a database.

Chapter 3

TEST DATABASES

In this work, three databases are used to test the performance of the proposed approach. These databases are referred to as:

- Natural Texture Mosaics Database I
- Natural Texture Mosaics Database II
- Natural Scenes Database

3.1 Natural Texture Mosaics Database I

The "Natural Texture Mosaics Database I" is a collection of images, which are mosaics of natural textures, [23]. This collection has 30 images, and all the images are variations of different degrees from a primary or seed image. The variations span across all aspects of the images; such as, slight changes in texture area, shape, and position, and changes in the textures used as well. This database is constructed to measure and investigate the level of sensitivity of the system and its associated algorithms to various degrees of variations in imagery presented to it. The images in this database are shown in Figure 1, where the first image at the top left corner of the first part of Figure 1 is the seed image.

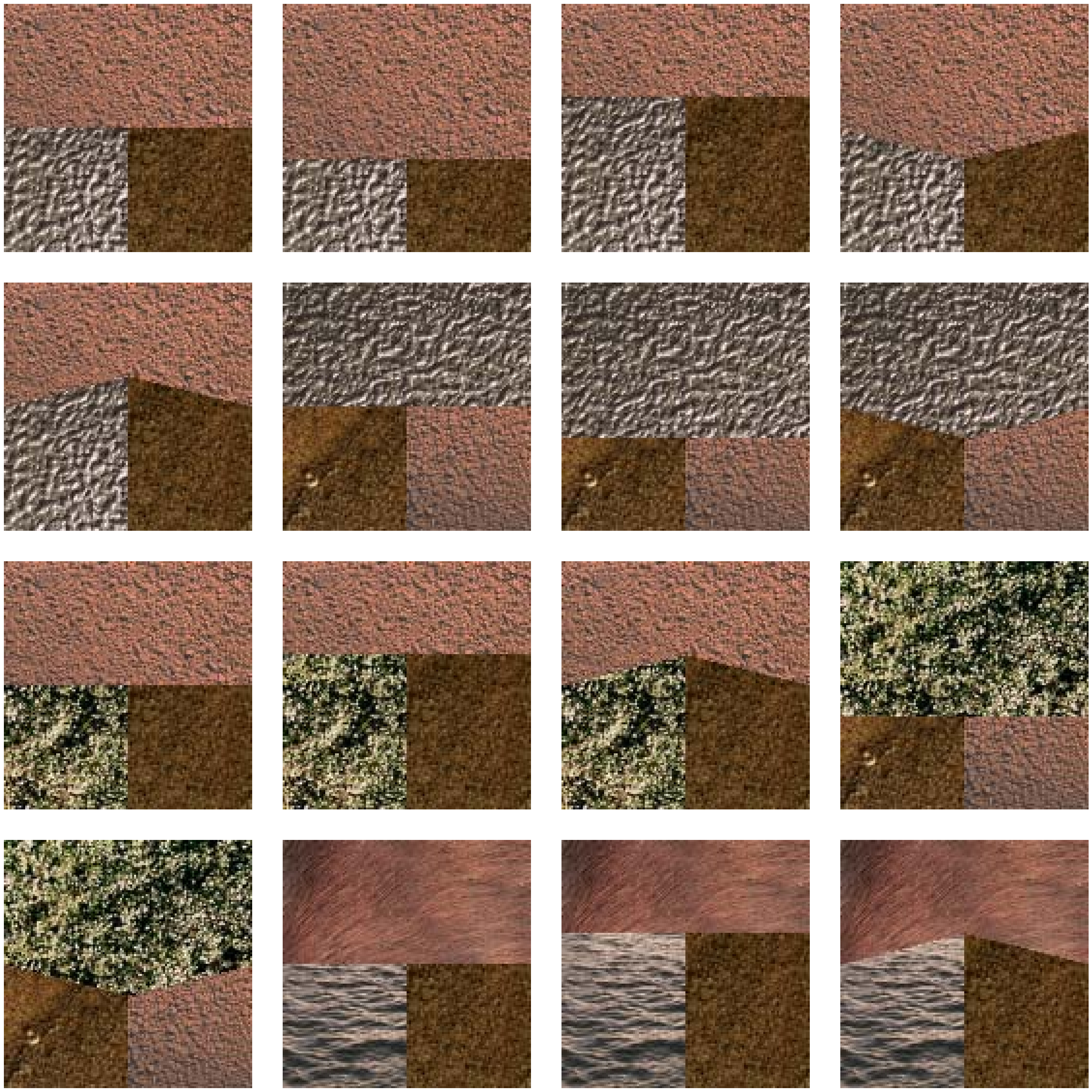


Figure 1. Images in the Natural Textures Mosaics Database I.

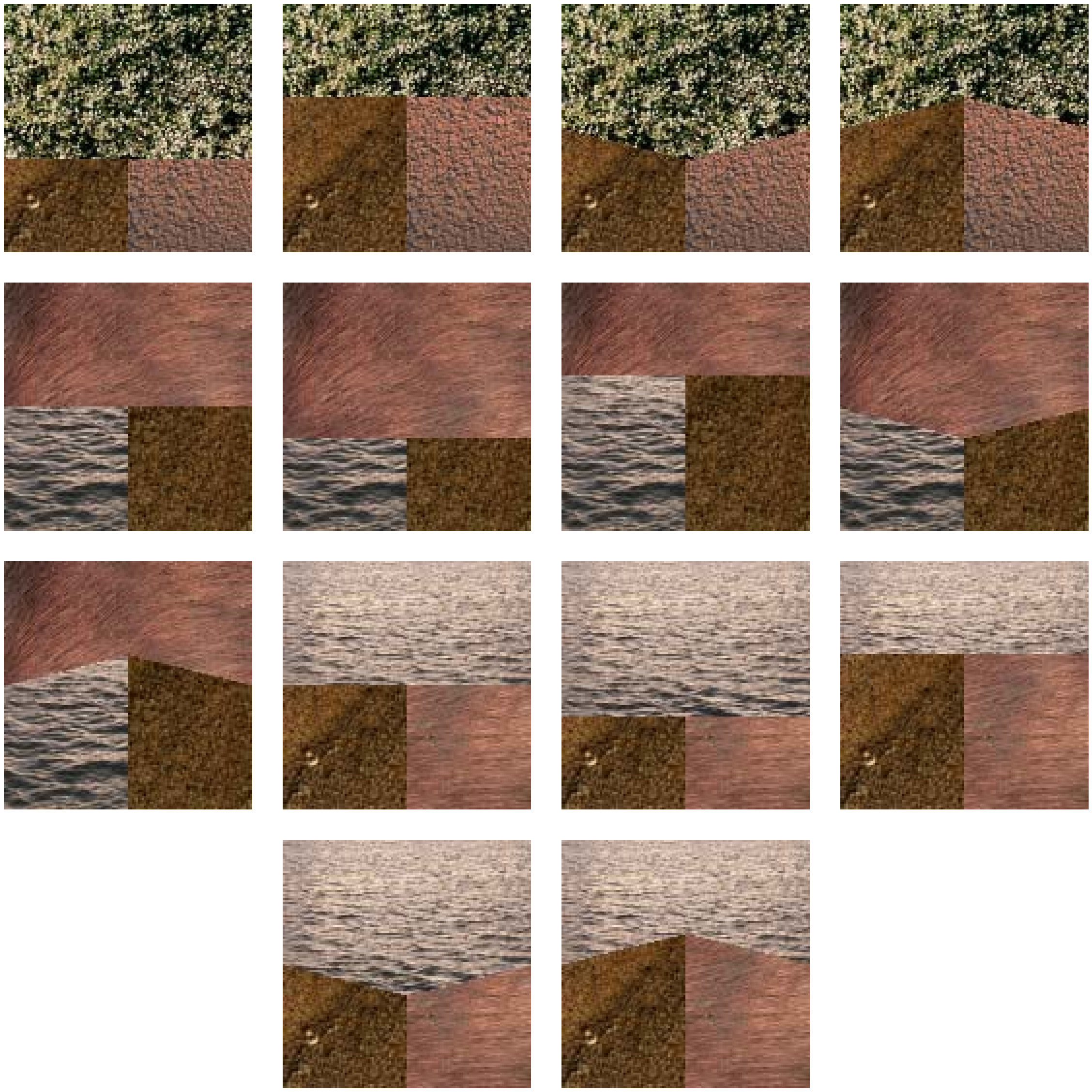


Figure 1 (continued). Images in the Natural Textures Mosaics Database I.

3.2 Natural Texture Mosaics Database II

The Natural Texture Mosaics Database II is a collection of images that are mosaics of natural textures constructed from texture images available in [23]. This collection includes 104 128 x 128 images with mosaics put together at random in terms of the arrangement and the type of the texture regions used. This database is constructed to measure and investigate the effectiveness of the algorithm in a controlled environment. Samples of images in this database are shown in Figure 2, and the complete database is shown in "APPENDIX A".

Figure 2. Samples of Images in the Natural Textures Mosaics Database II.

3.3 Natural Scenes Database

The Natural Scenes Database is a collection of images, which is comprised of natural scenes available in [24]. This collection has 313 images of various sizes that include 120 x 80, 80 x 120, 128 x 85, 85 x 128, 128 x 96, and 96 x 128 pixels. This database is constructed to ultimately test the performance on real imagery. Samples of images of this database are shown in Figure 3, and the complete database is shown in "APPENDIX B".

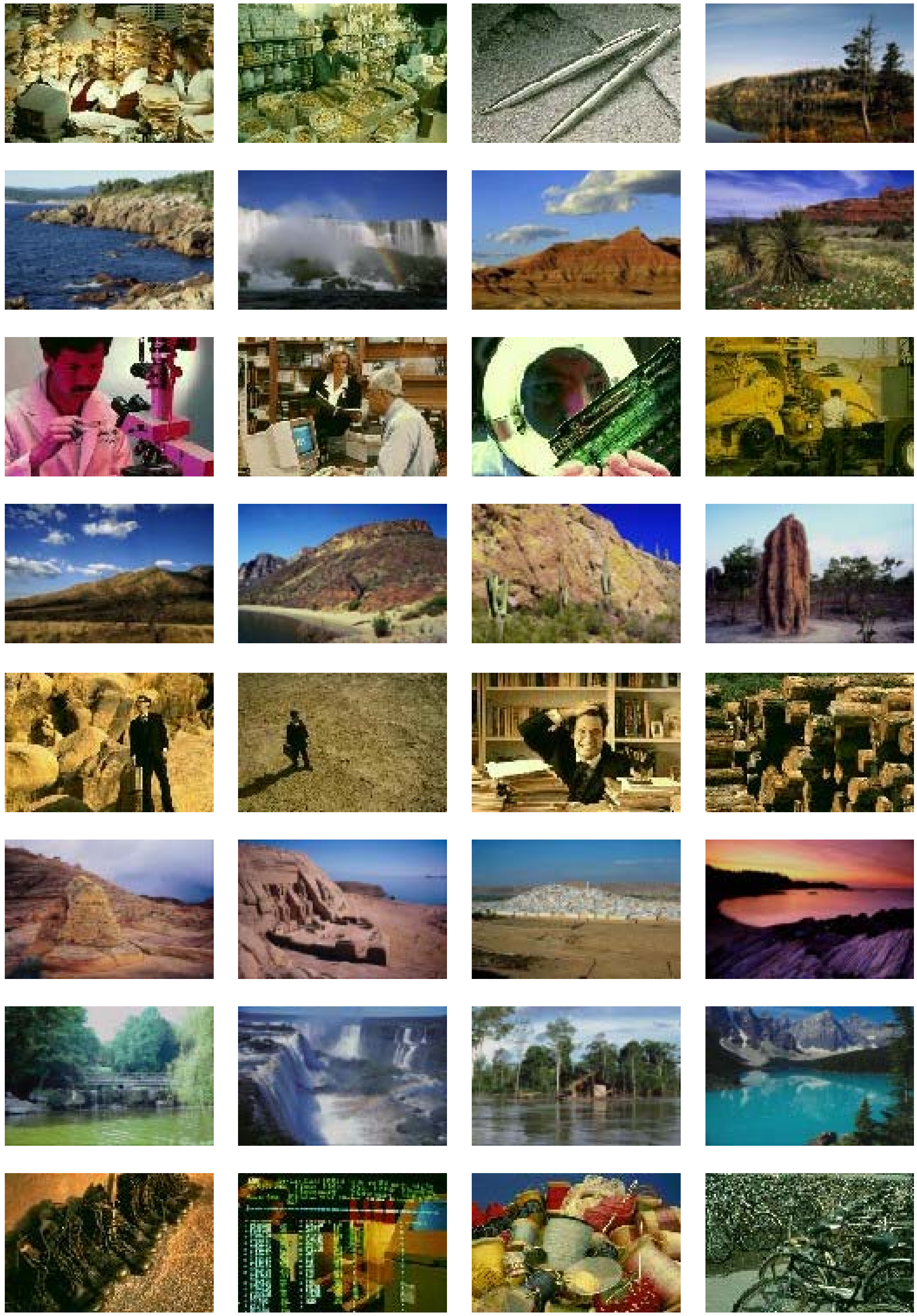


Figure 3. Samples of Images in the Natural Scenes Database.

3.4 Conclusions

In this chapter, the different databases that are used throughout this work are presented and explained. These databases have successfully fit the objective, as will be seen ahead in the rest of the body of this dissertation. These databases are used to produce the dissertation experimental results as shown later.

Chapter 4

TEXTURE CHARACTERIZATION BY COLOR TEXTURE AND COLOR CONTENT FEATURES

Characterizing color texture via a set of features is the subject of this chapter. Two types of textural features are utilized in this work. The first kind of features, which are based on random field models, characterize the interactions among inter and intra plane pixels. The second feature type focuses on the color space, and characterizes the pure color content without regard for textural activity. The combination of these two types of features is used for final color texture characterization.

4.1 Color Texture Characterization with Multispectral Simultaneous Autoregressive Model

In this work, the texture of the color images is characterized using a class of multispectral random field image model called the Multispectral Simultaneous Autoregressive (MSAR) model. The MSAR model is an extension of the gray-level Simultaneous Autoregressive (SAR) model, as first developed in [7], [25]. In these works, the MSAR model has been shown to be effective for color texture synthesis and classification [7], [25]. For mathematical simplicity, the model is formulated using a toroidal lattice assumption. A location within a two-dimensional M x M lattice is denoted by $\mathbf{s} = (i, j)$, with i, j being integers from the set $J = \{0, 1, \ldots, M\text{-}1\}$. The set of all lattice locations is defined as $\Omega = \{\mathbf{s} = (i, j) : i, j \in J\}$. The value of an image observation at location $\mathbf{s}$ is denoted by the vector value $\mathbf{y}(\mathbf{s})$, and the image observations are assumed to have zero mean. The MSAR model relates each lattice position to its neighboring pixels, both within and between image planes, according to the following model equation:

$$y_i(s) = \sum_{j=1}^{P} \sum_{r \in N_{ij}} \theta_{ij}(r)\, y_j(s \oplus r) + \sqrt{\rho_i}\; w_i(s), \quad i = 1 \ldots P$$

Equation 4-1

where:

$y_i(s)$ = pixel value at location s of the i^{th} plane

s and r = two dimensional lattices

P = number of image planes (for color images, P = 3, representing: Red, Green, and Blue planes)

N_{ij} = neighbor set relating pixels in plane i to neighbors in plane j (only interplane neighbor sets, i.e. $N_{ij,\, i \neq j}$, may include the (0,0) neighbor)

θ_{ij} = coefficients which define the dependence of $y_i(s)$ on the pixels in its neighbor set N_{ij}

ρ_i = noise variance of image plane i

$w_i(s)$ = i.i.d. random variables with zero mean and unit variance

$\oplus$ denotes modulo M addition in each index.

The parameters associated with the MSAR model are θ and ρ vectors which collectively characterize the spatial interaction between neighboring pixels within and between color planes. The θ vectors and the ratios of the components of the ρ vector are taken as the feature set $\mathbf{f_T}$ representing the underlying color texture of the image. The ratios of ρ vector components used are ρ_r/ρ_g and ρ_r/ρ_b, which are illumination invariant [25], as discussed later. Note that the r, g, and b subscripts denote the red, green, and blue planes, respectively.

4.1.1 Least Squares Estimation of MSAR Model Parameters

A least squares (LS) estimate of the MSAR model parameters is obtained by equating the observed pixel values of an image to the expected value of the model equations [25]. For the MSAR model this leads to P independent systems of M^2 equations:

$$y_i(\mathbf{s}) = E\{y_i(\mathbf{s}) \mid \boldsymbol{\theta}_i\}, \quad \mathbf{s} \in \boldsymbol{\Omega} \;\; and \;\; i = 1 \ldots P$$

Equation 4-2

$$y_i = \sum_{j=1}^{P} \sum_{\mathbf{r} \in N_{ij}} \theta_{ij}(\mathbf{r}) \, y_j(\mathbf{s} \oplus \mathbf{r}) = \mathbf{q}_i(\mathbf{s})^T \; \boldsymbol{\theta}_i$$

Equation 4-3

This, in turn, leads to the following estimates:

$$\hat{\theta}_i = \left[\sum_{\mathbf{s} \in \Omega} \mathbf{q}_i(\mathbf{s}) \, \mathbf{q}_i^T(\mathbf{s}) \right]^{-1} \left[\sum_{\mathbf{s} \in \Omega} \mathbf{q}_i(\mathbf{s}) \, y_i(\mathbf{s}) \right]$$

Equation 4-4

and

$$\hat{\rho}_i = \frac{1}{M^2} \sum_{\mathbf{s} \in \mathbf{\Omega}} \left(y_i(\mathbf{s}) - \hat{\boldsymbol{\theta}}_i^T \ \mathbf{q}_i(\mathbf{s}) \right)^2$$

Equation 4-5

where:

$$\boldsymbol{\theta}_i = [\boldsymbol{\theta}_{i1}^T \ \boldsymbol{\theta}_{i2}^T \cdots \boldsymbol{\theta}_{iP}^T]^T$$

Equation 4-6

$$\mathbf{q}_i(\mathbf{s}) = [\mathbf{y}_{i1}^T(\mathbf{s}) \ \mathbf{y}_{i2}^T(\mathbf{s}) \cdots \mathbf{y}_{iP}^T(\mathbf{s})]^T$$

Equation 4-7

$$\mathbf{y}_{ij}(\mathbf{s}) = col \ \{ y_j(\mathbf{s} \oplus \mathbf{r}) : \mathbf{r} \in N_{ij} \}$$

Equation 4-8

In this work, the neighborhood used for the MSAR model is a set that contains neighbors above, below, to the left, and to the right of the pixel as illustrated in Figure 4. The same neighbor set is used for both inter and intra-planes of the model.

```
            (0, 1)
              X

(-1, 0) X     O     X (1, 0)

              X
           (0, -1)
```

Figure 4. Neighbor Set Used with the MSAR Model.

This neighbor set results in a 20-dimensional $\mathbf{f}_T$. Therefore together with the two-dimensional color content feature set, a 22-dimensional C^3T feature vector, $\mathbf{f}$, is used to characterize each window described in Chapter 5.

4.1.2 MSAR Based Texture Synthesis

This section describes the effectiveness of this model for color texture synthesis. Figure 5 shows the neighborhood used, and Figure 6 shows the synthesis results. The top row shows the original images, and the bottom row includes the images synthesized from the MSAR least square estimates from the original image samples. The synthetic images can be visually compared to the original images, and observed to be very similar in appearance to the images from which the model was derived. Therefore, the $\mathbf{f}_T$ feature set does indeed capture the main characteristics of color textures.

		o	X	X
X	X	X	X	X
	X	X	X	

Figure 5. Neighbor Set Used for Image Synthesis Results in Figure 6.

Figure 6. MSAR Model Image Synthesis Results. Top Row: Original Color Texture Images. Bottom Row: Synthesized Images Using MSAR Parameters Estimated from Original Images.

4.2 Color Content Characterization

In addition to modeling color texture, the general color content of the image is also important. Therefore, in addition to the MSAR features, additional features focusing on the color alone are also considered. Two ways of characterizing the color content in the images using the sample mean of the color values (red, green, and blue) were investigated. These were the color means (a 3-dimensional feature vector), and the ratios of the color means (a 2-dimensional feature vector), which are illumination invariant. The ratios of the color means is selected as the feature of choice since extensive experimentation demonstrated its superiority over the color means. This feature set is defined as:

$$\mathbf{f}_C = \left\{ \frac{\hat{\mu}_r}{\hat{\mu}_g}, \frac{\hat{\mu}_r}{\hat{\mu}_b} \right\}$$

Equation 4-9

with $\hat{\mu}_i$ s being the sample mean of the color component of the red, green, and blue (r,g,b) planes respectively. Assuming that the observed value at each pixel is a product of illumination and spectral reflectance, the ratios of the color means are invariant to uniform changes in illumination intensity (i.e. the power of the illumination source changes uniformly across the spectrum). This kind of uniform change would cause each $\hat{\mu}_i$ to change by the same scale factor making the defined ratios invariant to illumination changes. This property makes the color-content features more robust. The same justification is used in the case of the ρ vector parameters of the MSAR model. To make the MSAR model illumination invariant, ρ_r/ρ_g and ρ_r/ρ_b are used in place of the ρ parameter of a single plane.

The combination of $\mathbf{f}_C$ and $\mathbf{f}_T$ features is used to represent a color texture region in this work. This combination is referred to as C^3T (Color Content, Color Texture) features from this point on, i.e., $C^3T = \{\mathbf{f}_C, \mathbf{f}_T\}$. Note that C^3T is a 22-dimensional vector as shown in Table 1.

Table 1. Components of the C^3T Feature Set.

ρ parameters	ρ_r/ρ_g
	ρ_r/ρ_b
Inter-Plane	$\theta_{rg}(1,0)$
	$\theta_{rg}(0,1)$
	$\theta_{rb}(1,0)$
	$\theta_{rb}(0,1)$
	$\theta_{gr}(1,0)$
	$\theta_{gr}(0,1)$
	$\theta_{gb}(1,0)$
	$\theta_{gb}(0,1)$
	$\theta_{br}(1,0)$
	$\theta_{br}(0,1)$
	$\theta_{bg}(1,0)$
	$\theta_{bg}(0,1)$
Intra-Plane	$\theta_{rr}(1,0)$
	$\theta_{rr}(0,1)$
	$\theta_{gg}(1,0)$
	$\theta_{gg}(0,1)$
	$\theta_{bb}(1,0)$
	$\theta_{bb}(0,1)$
Color Content	μ_r/μ_g
	μ_r/μ_b

4.3 Conclusions

In this chapter, the features used to characterize a color texture region are described. These features, referred to as C^3T features, are a combination of the MSAR features obtained via estimating the parameters of a MSAR model fitted to a region, and color-space based features. From this point on, the C^3T features are used to characterize a texture region.

Chapter 5

UNSUPERVISED SEGMENTATION WITH A HISTOGRAM-BASED CLUSTERING ALGORITHM

The first step in the proposed retrieval process is to segment the query image into regions of uniform color texture. It should be noted that since the ultimate goal is to retrieve images similar to the one presented to the system, it is necessary to find dominant regions of texture in the image but locating very exact boundaries of such regions is not as critical.

The segmentation algorithm used in this work relies on scanning the image with a sliding window and extracting C^3T features from each window. These features are then clustered using an unsupervised histogram-based algorithm. Mapping the identified clusters back into the image domain results in the desired segmentation.

5.1 Feature Extraction with a Sliding Window

The windowing operation consists of sliding a window from left to right and top to bottom across the image as illustrated in Figure 7. M is the size of the image in pixels, W is the size of the window in pixels, and D is the size of the sliding step in pixels. After extensive experimentation, where the value of D is varied from 1 pixel to W pixels, D is set to 4 pixels for this work, as this value yielded the best results. With D having a value of W pixels, the sliding windows are non overlapping and adjacent to each other. To find the optimum window size for each case, the size of the window W varies from 4 to 28 pixels in increments of 4 pixels. The best W is found automatically as described in later sections.

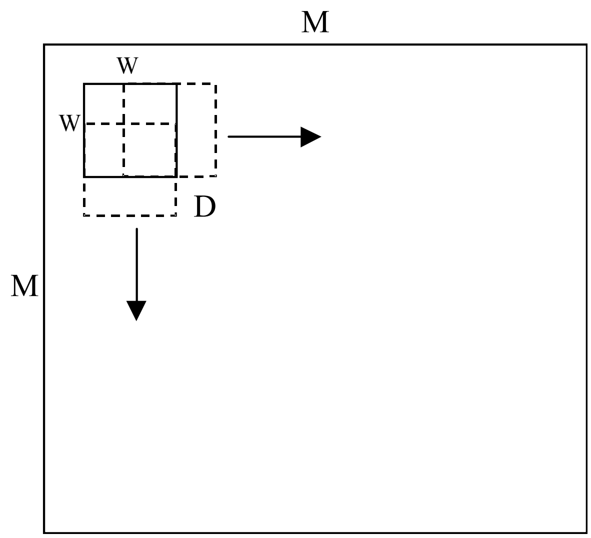


Figure 7. Feature Vector Extraction with a Sliding Window.

The texture bounded by each window is characterized using the 22-dimensional C^3T features.

5.2 Clustering Algorithm

Once all 22-dimensional C^3T features are extracted from the sliding window, they are clustered in the feature space using an unsupervised histogram-based peak climbing algorithm [26], [27], [28]. The 22-dimensional histogram is generated by quantizing each dimension according to the following:

$$CS(k) = \frac{f_{\max}(k) - f_{\min}(k)}{Q} \quad k = 1, 2, \ldots, N$$

Equation 5-1

$$d_k = INT\left\{ \frac{f(k) - f_{\min}(k)}{CS(k)} + 1 \right\} \quad k = 1, 2, \ldots, N$$

Equation 5-2

where:

N = total number of features (22 in this case)

CS(k) = length of the kth side of histogram cell

$f_{max}(k)$ = maximum value of the kth C^3T features

$f_{min}(k)$ = minimum value of the kth C^3T features

Q = total number of quantization levels

d_k = kth index for a histogram cell

Since the dynamic range of the vectors in each dimension can be quite different, the cell size for each dimension would be different. Hence the cells will be hyperboxes. Next, the number of feature vectors falling in each hyperbox is counted and this count is associated with the respective hyperbox creating the required histogram.

After the histogram is generated in the feature space, a peak climbing clustering approach is utilized to group the features into distinct clusters. This is done by locating the peaks of the histogram. In Figure 8 this peak climbing approach is illustrated for a two-dimensional space example.

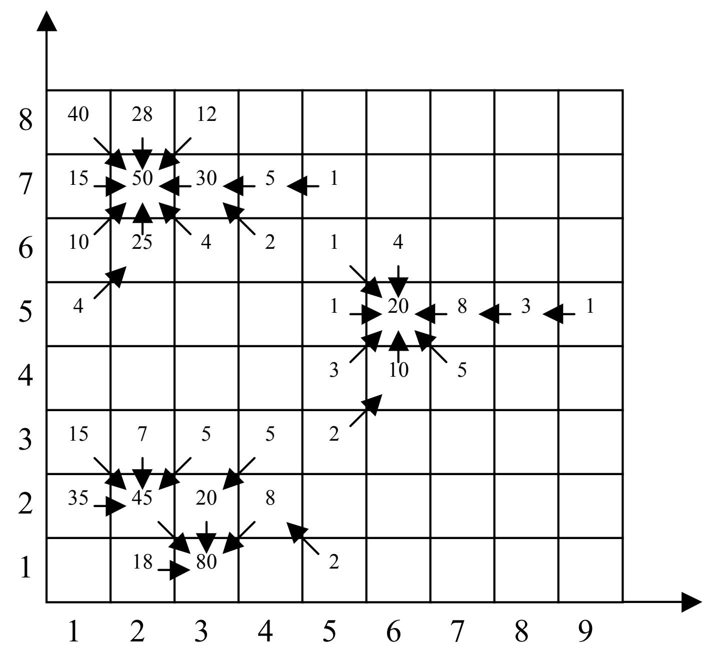


Figure 8. Illustration of the Peak Climbing Approach for a Two-Dimensional Feature Space Example.

The number in each cell (hyperbox) represents a hypothetical count for the feature vectors captured by that cell. By examining the counts of the 8-neighbors of a particular cell, a link is established between that cell and the closest cell having the largest count in the neighborhood. At the end of the link assignment, each cell is linked to one parent cell, but can be parent of more than one cell. A peak is defined as being a cell with the largest density in the neighborhood, i.e. a cell with no parent. A peak and all the cells that are linked to it are taken as a distinct cluster representing a mode in the histogram. Once the clusters are found, the windows associated with features grouped in the same cluster are tagged as belonging to the same category.

5.2.1 Automatic Selection of Quantization Level, Cluster Validation, and Automatic Selection of Window Size

A major component of this algorithm is the number of quantization levels associated with each dimension. To decide this parameter, the total number of non-empty cells and the percentage of them capturing only one vector for each selection of quantization levels are examined. The best number of quantization levels is selected as the largest one that maximizes the measure below [28].

$$M_i = (N_{ci} - N_{ui}) \times N_{ui}$$

Equation 5-3

where:

N_{ci} = number of non-empty cells

N_{ui} = number of cells capturing only one sample

The algorithm also includes a spatial domain cluster validation step. This step involves constructing a matrix B for each cluster m as:

$$\begin{aligned} B_m(i, j) &= 1 \quad \textit{if sample} \in \textit{cluster } m \\ B_m(i, j) &= 0 \quad \textit{otherwise} \end{aligned}$$

Equation 5-4

The (i, j) index corresponds to the location of a sliding window. A cluster is considered compact if only a very small number of its 1-elements have a 0-element neighbor, i.e. a cluster is considered valid (compact) if only a very small number of its elements have neighboring elements that do not belong to that cluster. A cluster that does not pass this test is merged with a valid cluster that has the closest centroid to it.

The final step in the clustering algorithm is to merge small non-connected areas that are sub-components of a bigger texture area in the spatial domain. These small areas are merged with the connected component area in the image that has the closest centroid if they are less than 2% (area-wise) of the entire image.

During the segmentation process, the best window size for scanning the image is chosen in an unsupervised fashion. The optimum window size is obtained by sweeping the image with varying window sizes (4 to 28 pixels in steps of 4 pixels), and choosing the smallest one out of at least two consecutive window sizes that produce the same number of clusters.

The last step in the segmentation algorithm is to map all the image windows in a particular cluster back to the spatial domain of the image and mark them as belonging to the same texture. The resulting image contains regions that are identified as distinct regions, and as such further processing on them can be done. Figure 9 illustrates this step. Note that in the segmented image, the top and bottom textures, which are the same, receive the same gray level value.

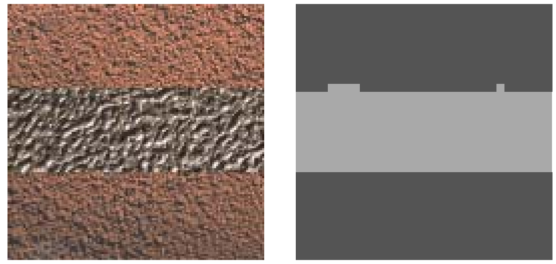


Figure 9. Illustration of Color Texture Image Segmented into Regions.

5.3 Experimental Results

The performance of the proposed segmentation algorithm and the associated features is illustrated in Figure 10 and Figure 11. Figure 10 shows four images of the "Natural Textures Mosaics Database II", each containing a number of different textures. Below each image the segmentation result is presented in the form a gray-level image with pixels belonging to the same texture having the same gray level. In the next row, the boundaries of the segmented regions are shown as superimposed white lines. At the top of the image, the size of the optimal window found by the algorithm is also shown. Figure 11 shows the segmentation results for several natural scene images from the "Natural Scenes Database". It is observed that the proposed algorithm performs quite well and is capable of localizing uniform color textures in each image.

In Figure 10, Figure 11, and Figure 12 the results of the approach used in this work are compared with that of the image segmentation results achieved using the JSEG method described by Deng and Manjunath in [7]. The JSEG method is an unsupervised segmentation approach for color-texture regions in images. It consists of two independent steps: color quantization and spatial segmentation. In the first step, colors in the image are quantized to several representative classes that can be used to differentiate regions in the image. Applying the criterion to local windows in the class-map results in the "*J*-image," in which high and low values correspond to possible boundaries and interiors of color-texture regions. A region growing method is then used to segment the images based on the multiscale *J*-images. It should be noted that this method is pixel based.

The JSEG results are obtained by applying the images to the programs made available by the JSEG authors on the World Wide Web (WWW) site http://maya.ece.ucsb.edu/JSEG/. The obtained region boundaries are superimposed on the original images. The JSEG results are displayed in the last row of Figure 10, Figure 11, and Figure 12. It can be seen that the segmentation results using the method proposed in this work have a better match with perceptual boundaries in the images. The JSEG method over segments most of the natural scene images and misses or mislabels some boundaries in mosaic images.

Figure 10. Segmentation Results for Four Images from the "Natural Textures Mosaic Database II". 1st row: Original Image. 2nd row: Segmentation Results. 3rd row: Texture Boundaries Corresponding to Segmentation Results. 4th row: Segmentation Using the JSEG Method.

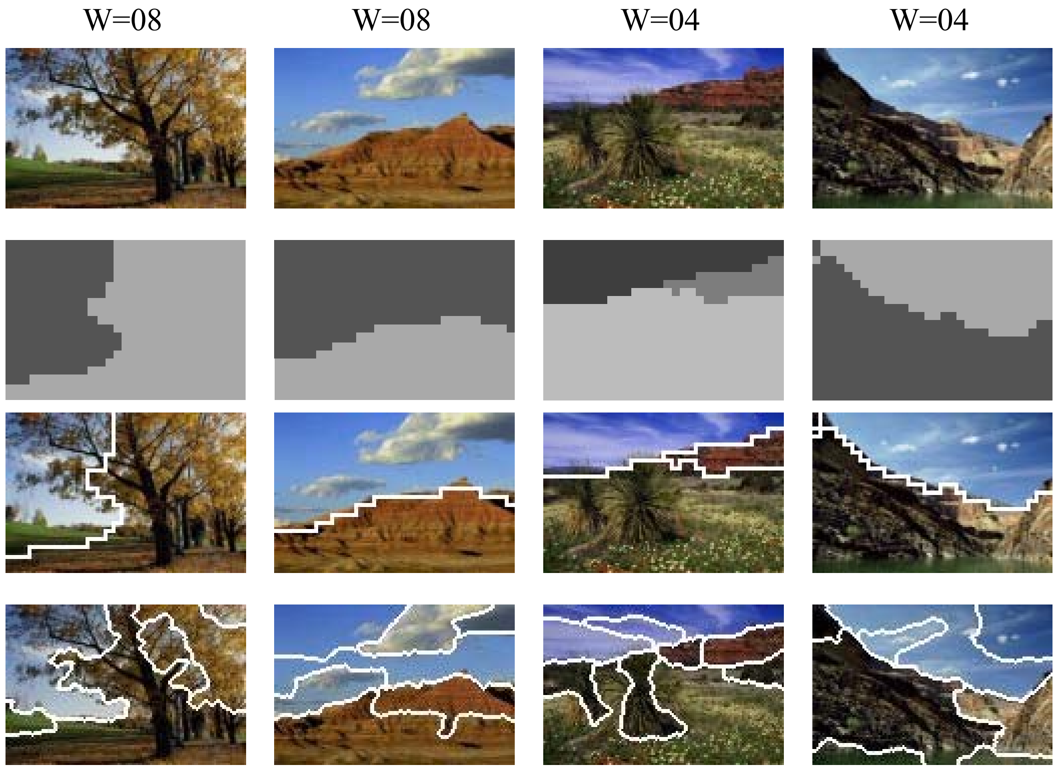


Figure 11. Segmentation Results for Four Images from the "Natural Scenes Database". 1st row: Original Image. 2nd row: Segmentation Results. 3rd row: Texture Boundaries Corresponding to Segmentation Results. 4th row: Segmentation Using the JSEG Method.

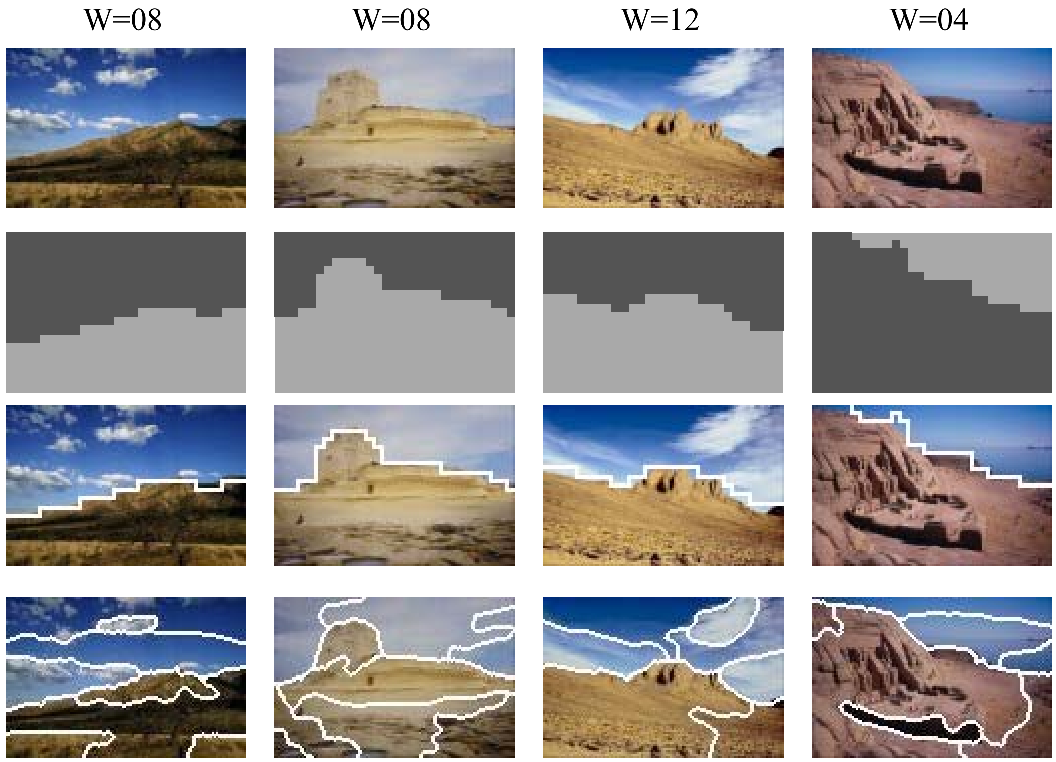


Figure 12. Segmentation Results for Another Set of Four Images from the "Natural Scenes Database". 1st row: Original Image. 2nd row: Segmentation Results. 3rd row: Texture Boundaries Corresponding to Segmentation Results. 4th row: Segmentation Using the JSEG Method.

In Figure 13, the result of the segmentation procedure on another real image (natural scene) is shown. Note that the people in the beach are merged with the sand, because they are approximately the same color and texture as the sand. The window size that is selected automatically as the best window size to perform the image segmentation is W = 4.

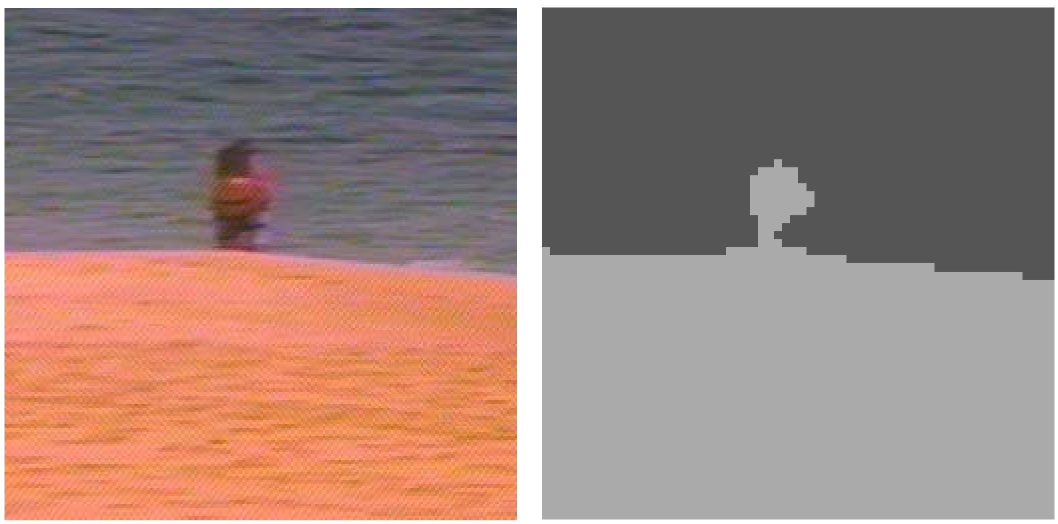


Figure 13. Segmentation Result for a Natural Scene.

5.4 Conclusions

In this chapter, the segmentation algorithm used in this work has been described. The approach uses unsupervised clustering to group similar C^3T features from the windows in the image being segmented, and then maps those clusters back to the spatial domain to yield the segmented image result. This approach constitutes a new method for image segmentation, and experimental results confirm the validity of the proposed method.

Chapter 6

ATTRIBUTES USED IN SIMILARITY METRICS

A key element in image retrieval is the definition of the similarity metrics used to measure the closeness of images. Since this work emphasizes color texture regions in the image, the developed similarity metrics concentrate on the characteristics associated with such regions. The type, geometric shape, and spatial arrangement of regions are the attributes that are taken into account. These attributes are described in this chapter.

6.1 Image Separation into Regions

Segmentation results in identifying distinct textures and the portions of an image associated with them. However, there may be non-connected parts of the image that contain similar texture as shown in Figure 14.

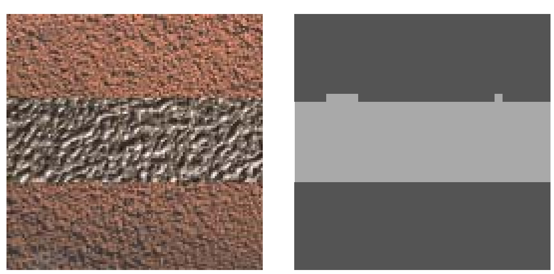


Figure 14. Example of an Image with Two Textures but Three Regions.

The well-known "Connected Component Labeling" [29] algorithm is applied to identify the number of connected regions regardless of their texture type. For the example shown in Figure 14, there are two textures from the segmentation results, but three regions. All subsequent processing is done on these regions.

6.2 Metrics Description

Once the images are segmented into regions of distinct color texture, similarity metrics need to be developed to measure how close two images are with respect to their color texture content. The attributes used for similarity computation are described in this section.

6.2.1 Texture-Based Metrics

The foremost attribute is the type of color texture of each segmented region. To characterize this texture, the largest square that could be fitted to the region is found. This square will be referred to as the "Maximum Fitting Square" (MFS). The C^3T features are then extracted from the MFS and used to characterize the color texture of the entire segmented region. The reason for using the MFS is that the shape of the segmented region could be irregular which will not be suitable for the MSAR model computation.

6.2.2 Shape-Based Metrics

Several shape-related parameters are then extracted from the entire region (not the MFS). These are:

1. Centroid location, which is the mean of the row and column positions of the pixels contained in the region, i.e. ($\bar{x}$, $\bar{y}$). The centroid location conveys information about the overall position of the region.
2. Area (A) which is the total number of pixels contained by the region.
3. Overall shape as measured by the moments that define the best fitting ellipse to a region, i.e. the largest ellipse that could be fitted to the region. Such an ellipse could be found by computing three geometrical moments, μ_{xx}, μ_{yy}, and μ_{xy} where:

$$\mu_{xx} = \sum_{\substack{OVER \\ ROWS}} \sum_{\substack{OVER \\ COLUMNS}} \frac{(x-\bar{x})^2}{A}$$

Equation 6-1

$$\mu_{yy} = \sum_{\substack{OVER \\ ROWS}} \sum_{\substack{OVER \\ COLUMNS}} \frac{(y-\bar{y})^2}{A}$$

Equation 6-2

$$\mu_{xy} = \mu_{yx} = \sum_{\substack{OVER \\ ROWS}} \sum_{\substack{OVER \\ COLUMNS}} \frac{(x-\bar{x})\,(y-\bar{y})}{A}$$

Equation 6-3

These three shape-based measures along with the C^3T features extracted from the MFS are used in the similarity metric developed for the retrieval process.

Figure 15 and Figure 16 illustrate this process for three texture mosaics and three natural scene images, respectively. In each figure, the original images are shown in the first column followed by the computed MFS' superimposed on the segmented regions.

The centroid locations, and the best fitting ellipses in images are shown in columns three and four, respectively.

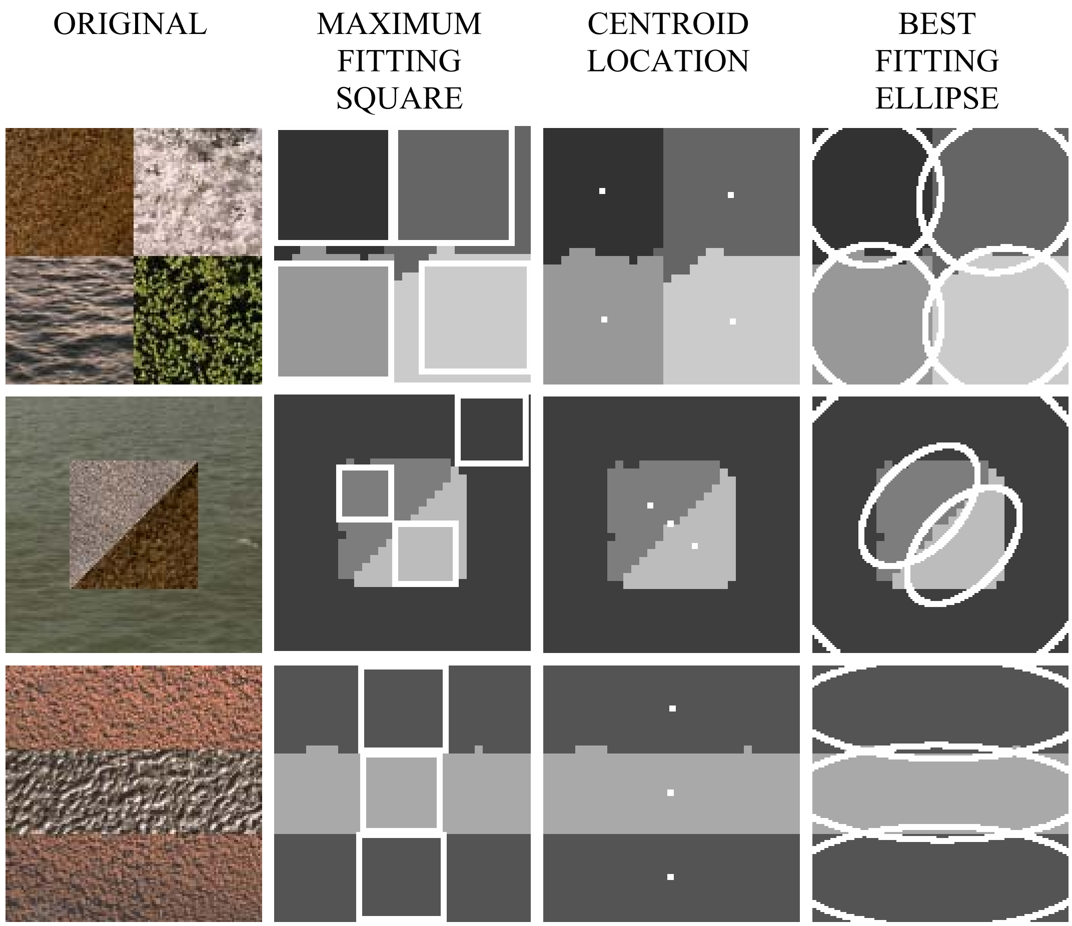


Figure 15. Examples of Texture and Shape Attribute Computation for Texture Mosaic Images.

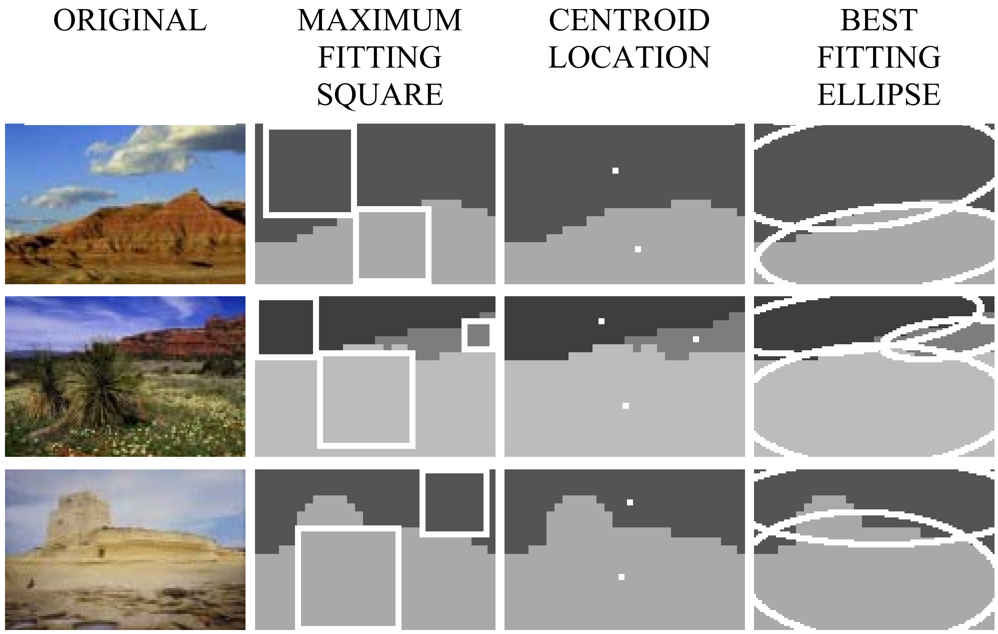


Figure 16. Examples of Texture and Shape Attribute Computation for Natural Scenes.

Additionally, Figure 17 and Table 2 together show numerical details for metrics that are obtained through the methodology described previously.

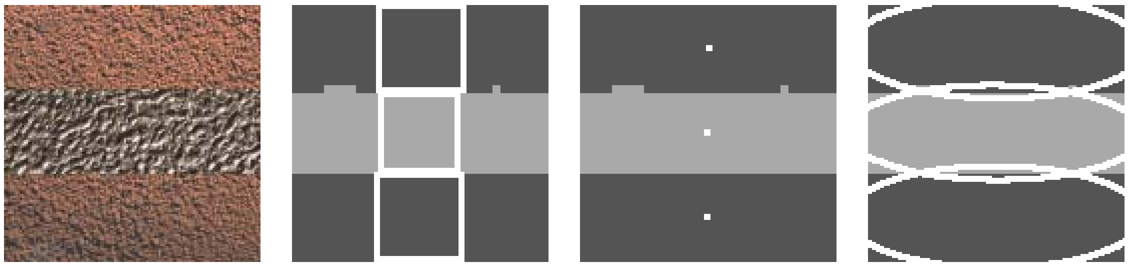


Figure 17. Image for Characterization Parameters in Table 2.

Table 2. Sample of Characterization Parameters for Figure 17.

Parameter	Top Texture	Middle Texture	Bottom Texture
Area	5552	5200	5632
Centroid			
$\bar{x}$	64	63	63
$\bar{y}$	21	63	105
Moments			
μ_{xx}	1361.966859	1368.521538	1365.500000
μ_{yy}	157.753602	138.576923	161.500000
μ_{xy}	7.101585	8.151538	0.250000
Color & Texture			
μ_r/μ_g	1.439413	1.119362	1.351956
μ_r/μ_b	1.762133	1.242700	1.599600
ρ_r/ρ_g	2.603568	1.076903	2.666817
ρ_r/ρ_b	4.133424	1.095415	4.966406
$\theta_{rr}(1,0)$	0.444628	0.971350	0.823327
$\theta_{rr}(0,1)$	0.719858	0.995408	0.920288
$\theta_{rg}(1,0)$	-0.388314	-0.965591	-1.152890
$\theta_{rg}(0,1)$	-0.877187	-1.071018	-0.912498
$\theta_{rb}(1,0)$	-0.341331	0.236650	-0.137124
$\theta_{rb}(0,1)$	0.241477	0.352867	-0.404037
$\theta_{gr}(1,0)$	0.278831	0.897633	0.452329
$\theta_{gr}(0,1)$	0.177190	0.790400	0.450098

$\theta_{gg}(1,0)$	-0.338305	-1.010049	-0.691215
$\theta_{gg}(0,1)$	-0.095870	-0.879615	-0.430971
$\theta_{gb}(1,0)$	-0.138935	0.339629	-0.042100
$\theta_{gb}(0,1)$	0.102538	0.354909	-0.221388
$\theta_{br}(1,0)$	0.263438	0.952105	0.378532
$\theta_{br}(0,1)$	0.094938	0.790305	0.351593
$\theta_{bg}(1,0)$	-0.527696	-1.425158	-0.797446
$\theta_{bg}(0,1)$	-0.217736	-1.118033	-0.574633
$\theta_{bb}(1,0)$	0.106645	0.698588	0.229757
$\theta_{bb}(0,1)$	0.336770	0.591262	0.105285

6.3 Conclusions

In this chapter, details of the attributes extracted from the segmented images to be used in the similarity metrics computation has been described. The distinction between textures and regions to extracts this attributes is explained as well. Both, the texture based and shape based attributes, are described in detail, and examples are presented. Collectively, these attributes represent a novel approach to abstracting the content of a color texture image to a very condensed hight level meta-data description in a text file.

Chapter 7

THE RETRIEVAL PROCESS

This chapter describes the image retrieval process and the steps that are associated with it. These include:

1) Search Space Reduction:	The process of narrowing the search to images that contain similar textures to the query image.
2) Region Association:	The algorithm that associates each region of the query image to one of the regions of a database image.
3) Similarity Computation:	The definition of the similarity measure and its computation.
4) Final Retrieval:	The retrieval process based on the computed similarity measure.

7.1 Search Space Reduction

The next step after extracting the texture and shape attributes is to reduce the number of database images that are to be considered in the retrieval process. This is done by finding those images that contain similar textures to that of the query image. This task is carried out by using the C^3T features extracted from the MFS. The C^3T features of the database images are first clustered using the clustering algorithm described earlier resulting in M clusters, C_i, $i = 1, 2, \ldots, M$. Then, the C^3T features of each of the segmented regions of the query image are considered in this clustered 22-dimensional space. Let's denote the C^3T features of the kth segmented region of the query image as $\mathbf{f}_k$. For each $\mathbf{f}_k$, the closest cluster center $C_j(\mathbf{f}_k)$ is found and all the database images that are associated with the C_j cluster are tagged as images that need to be considered in the search process. After all $\mathbf{f}_k$'s are considered, all the database images that are not tagged are removed from further consideration resulting in a reduced search space. In other words, only those database images that potentially have one or more similar color texture to those of the query image are retained. Note that this process also identifies the likely number of common textures between the query image and each of the retained database images.

The images in the reduced search space are ranked with respect to the initial query image. These images are grouped and ranked starting with the ones that have the most number of common textures with the query image, denoted by T, and ending with the ones that have only 1 texture of the query image. This is done in the sequence/rank: T, T-1, T-2, …, 3, 2, 1.

For each group above, the region association, similarity computation, and retrieval processes are carried out (from group whose images contain at least T textures of the query image to group whose images contain at least 1 texture of the query image or to the group that contains the x^{th} image - whichever happens first). The x^{th} image is a parameter passed to the algorithm that indicates that the x most similar images to the query image are to be retrieved from the database. The C^3T features from the MFS of the regions, and all the attributes discussed in Chapter 6 have different dynamic range and magnitudes. Since they are to be used jointly in the retrieval process, their effect should be equalized. This is done by normalizing the C^3T features from the MFS of the regions, and each geometrical attribute to have zero mean and unit variance through subtracting the sample mean and dividing by the sample standard deviation. The sample mean and standard deviation are calculated for the query image and each group of image databases that have the same number of common textures with the query.

7.2 Region Association

In this phase, the retained database images are considered one at a time and an association is established between each region of the query image and one of the regions of the considered database image. This process is illustrated by an example.

Let's assume that the query image contains three segmented regions, Q_1, Q_2, and Q_3 and a database image has two regions denoted by R_1 and R_2 as shown in Figure 18.

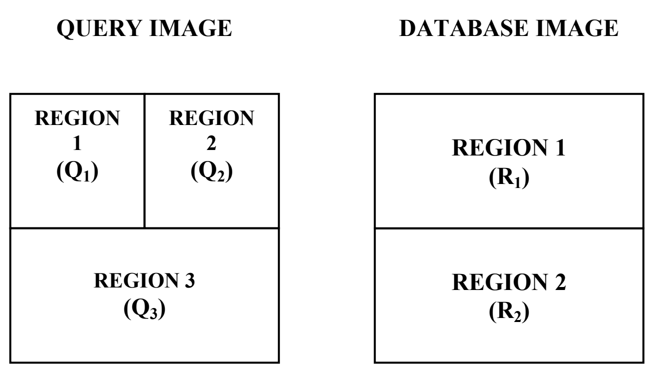


Figure 18. An Example of the Region Association Process.

The Euclidean distance between C^3T features of each possible Q_i, R_j pair ($d(Q_i, R_j)$) is computed resulting in the following six distances:

{$d(Q_1, R_1)$, $d(Q_1, R_2)$, $d(Q_2, R_1)$ and $d(Q_2, R_2)$, $d(Q_3, R_1)$ and $d(Q_3, R_2)$}

These six distances are then sorted from smallest to largest. Let's assume this sorting results in the following order:

{$d(Q_3, R_1)$, $d(Q_2, R_1)$, $d(Q_3, R_2)$, $d(Q_1, R_2)$, $d(Q_1, R_1)$, $d(Q_2, R_2)$}

An association between each Q_i and R_j is established by considering this sorted list and making the association according to the first occurrence of each Q_i in the list. In this example, the following associations are made:

$Q_3 \leftrightarrow R_1$ $\quad$ $Q_2 \leftrightarrow R_1$ $\quad$ $Q_1 \leftrightarrow R_2$

Upon completion of this step, the best potential match between each region of the query image and one of the regions of each of the database images is established.

7.3 Similarity Computation

The similarity between the query image and a database image is computed using the region associations established in the previous phase. The texture based C^3T features are already used to identify the similar regions between the two images. In this stage, the shape-based attributes are utilized to arrive at a final similarity measure. The Euclidean distances between the position, area, and shape attributes of each associated pair are computed. Continuing with the example from the previous section, the computed distances will be:

Q_3 associated with R_1 : $\quad d(P_{Q_3}, P_{R_1})$, $d(A_{Q_3}, A_{R_1})$, $d(S_{Q_3}, S_{R_1})$

Q_2 associated with R_1 : $d(P_{Q_2}, P_{R_1})$, $d(A_{Q_2}, A_{R_1})$, $d(S_{Q_2}, S_{R_1})$

Q_1 associated with R_2 : $d(P_{Q_1}, P_{R_2})$, $d(A_{Q_1}, A_{R_2})$, $d(S_{Q_1}, S_{R_2})$

where:

$d(P_{Q_i}, P_{R_j})$ = Euclidean distance between centroid locations of Q_i and R_j

Equation 7-1

$d(A_{Q_i}, A_{R_j})$ = Euclidean distance between areas of Q_i and R_j

Equation 7-2

$d(S_{Q_i}, S_{R_j})$ = Euclidean distance between 3-dimensional vector of (μ_{xx}, μ_{yy}, μ_{xy}) of Q_i and R_j

Equation 7-3

Next, the position, area, and shape distances for all Q_i's are added together to form total position, area, and shape difference measures denoted by PD, AD, and SD, respectively.

$$PD = d(P_{Q_3}, P_{R_1}) + d(P_{Q_2}, P_{R_1}) + d(P_{Q_1}, P_{R_2})$$

Equation 7-4

$$AD = d(A_{Q_3}, A_{R_1}) + d(A_{Q_2}, A_{R_1}) + d(A_{Q_1}, A_{R_2})$$

Equation 7-5

$$SD = d(S_{Q_3}, S_{R_1}) + d(S_{Q_2}, S_{R_1}) + d(S_{Q_1}, S_{R_2})$$

Equation 7-6

At this point a total texture difference measure based on the Euclidean distance of the C^3T features is also computed and denoted as TD

$$\mathrm{TD} = d(C^3T_{Q_3}, C^3T_{R_1}) + d(C^3T_{Q_2}, C^3T_{R_1}) + d(C^3T_{Q_1}, C^3T_{R_2})$$

Equation 7-7

Finally a composite distance measure used as the overall similarity measure between the two images is computed. This measure denoted as S is:

$$S = \sqrt{PD^2 + AD^2 + SD^2 + TD^2}$$

Equation 7-8

The S measure serves as the final closeness measure between the query image and a database image. Its should also be noted that the terms of Equation 7-8 have the same dynamic range due to the normalization process described in section 7.1.

7.4 Retrieval

The S measure reflects the degree of similarity between the query image and the considered database image. As such, the retrieval process involves sorting all the S measures computed between the query image and each of the database images in the reduced search space. The database images ordered from smallest to largest S will be the most similar to least similar to the query image, respectively.

The performance of the system on two different databases is reported in the next chapter. In each experiment, the query image is taken to be one of the database images, i.e., there is an exact copy of the query image in the database to be searched. In each case, the exact copy is always found as the most similar image. This is an important affirmation of the validity of the proposed algorithm.

7.4.1 Overall System Description and Retrieval Algorithm

The flow chart in Figure 19 describes how the overall system works. First, there is an off line process to characterize the database. In this process, each image in an image database is segmented, and characterized. This means that for each region in the segmented image, the area, centroid, moments, and C^3T parameters are extracted, and included into the index ASCII file that each image file in the database has. After the database has been characterized, the on line process consists of characterizing the query image like it is done for the images in the database, determining the "Reduced Search Space", and calculating the similarity metric S for each of the images from the database that end up being members of the "Reduced Search Space". These images are ranked according to the similarity metric, and copied from the database directory to a directory that contains the retrieval results; this is the output directory from the query operation. During this ranking and output operation, a lower numerical value of the similarity metrics means that the image in the database is more similar to the query image. The next chapter describes the results with the test databases.

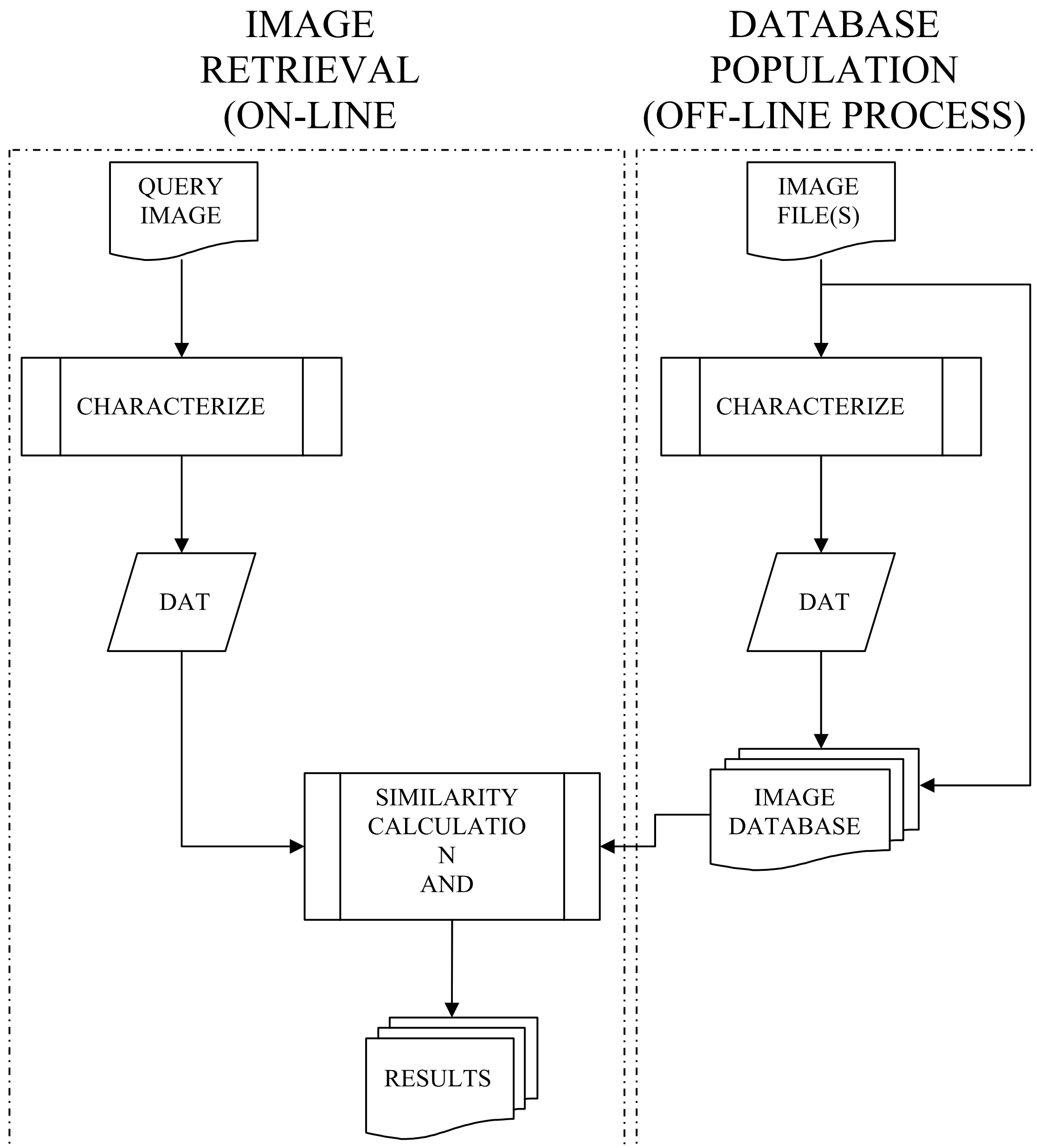


Figure 19. Overall System Block Diagram.

7.5 Conclusions

In this chapter, the various steps of the proposed retrieval process are described. This process involves: search space reduction, region association between a query and a database image, similarity computation, and ranking based on the computed similarity measures. A novel similarity metric that considers both textural and geometric features of a region is proposed. The entire process is automatic from start to finish, and the only input is the query image.

Chapter 8

EXPERIMENTAL RESULTS

This chapter includes the experimental results for different steps of the retrieval process. The three databases described in Chapter 3 are used in the experiments, and the images used as query in all the experimental results shown in this chapter were selected at random from the respective databases. The performance of the proposed retrieval system is documented, and issues related with the speed of execution are discussed.

8.1 Experimental Results for the Search Space Reduction

This section shows the experimental results for the "Search Space Reduction" step of the overall retrieval algorithm.

8.1.1 Experiments using a Natural Textures Database

The image with the blue border in Figure 20 is the query image. The rest of the images in the figure are selected from the database by the algorithm for containing similar textures to the query image.

Figure 20. Search Space Reduction Results for a Natural Texture Mosaics Database.

8.1.2 Experiments using a Natural Scenes Database

The image with the blue border in Figure 21 below is the query image. The rest of the images in the figure are selected from the database by the algorithm for containing similar textures to the query image.

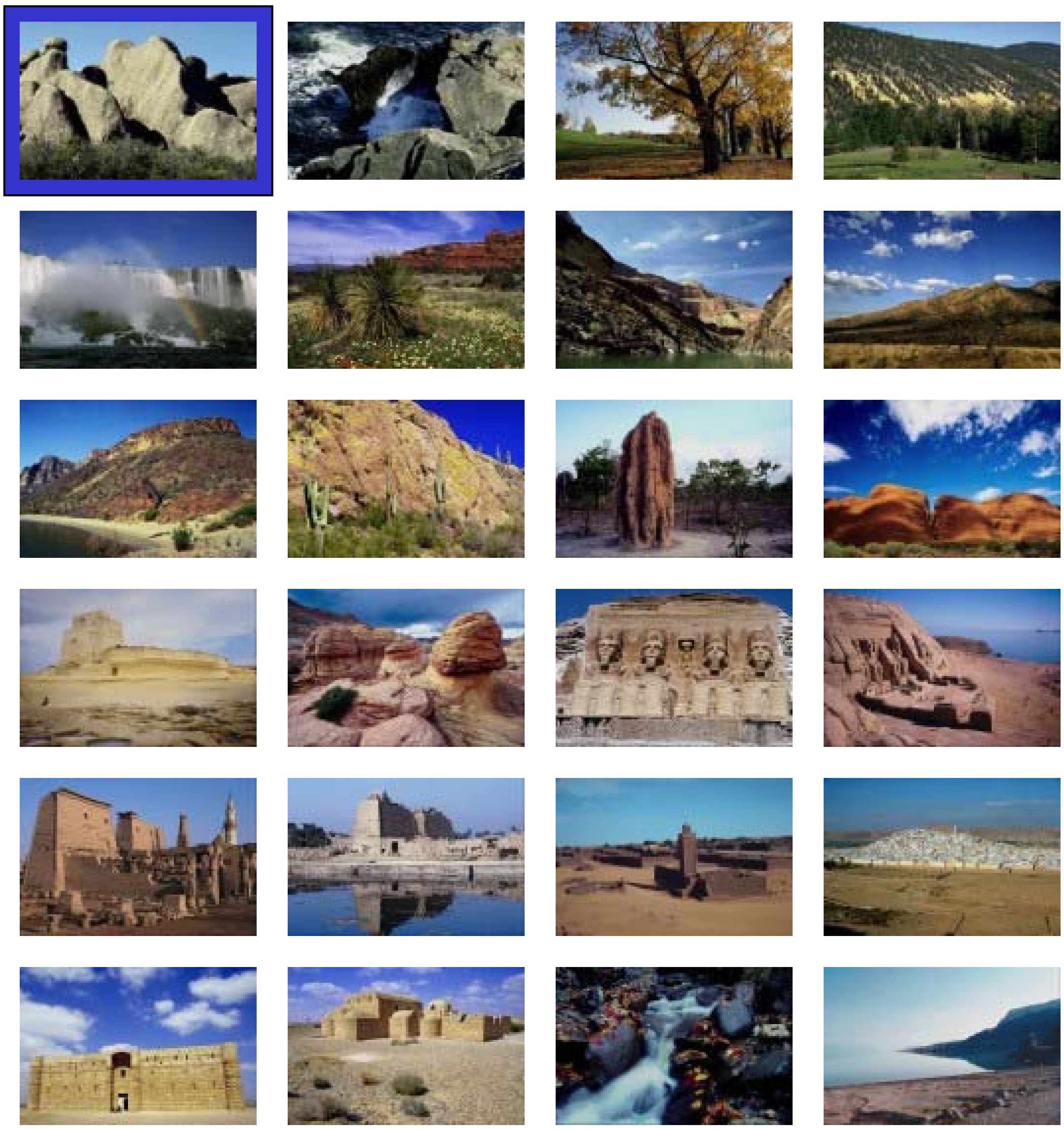


Figure 21. Search Space Reduction Results for a Natural Scenes Database.

8.2 Retrieval Results

A series of experiments are carried out by taking an image from the considered database, and using it as the query image. In all the experiments the database image that is the same as the query image is systematically found, matched, and ranked as the most similar image to the query image. The retrieved images are the bottom two rows of images in each figure. After each of the figures showing the results, an accompanying table that documents the final S value for each of the retrieved images is presented.

8.2.1 Experiments using Natural Texture Mosaics Database I

In Figure 22, the image with a blue border is the query image. The rest of the images in this figure are selected from the database (30 images) by the algorithm for being the most similar images to the query image. They are also ranked from most similar to least similar from left to right and top to bottom. Table 3 lists the corresponding S values for these images. The image in the database that is exactly the same as the query image is retrieved and ranked as the most similar to the query image.

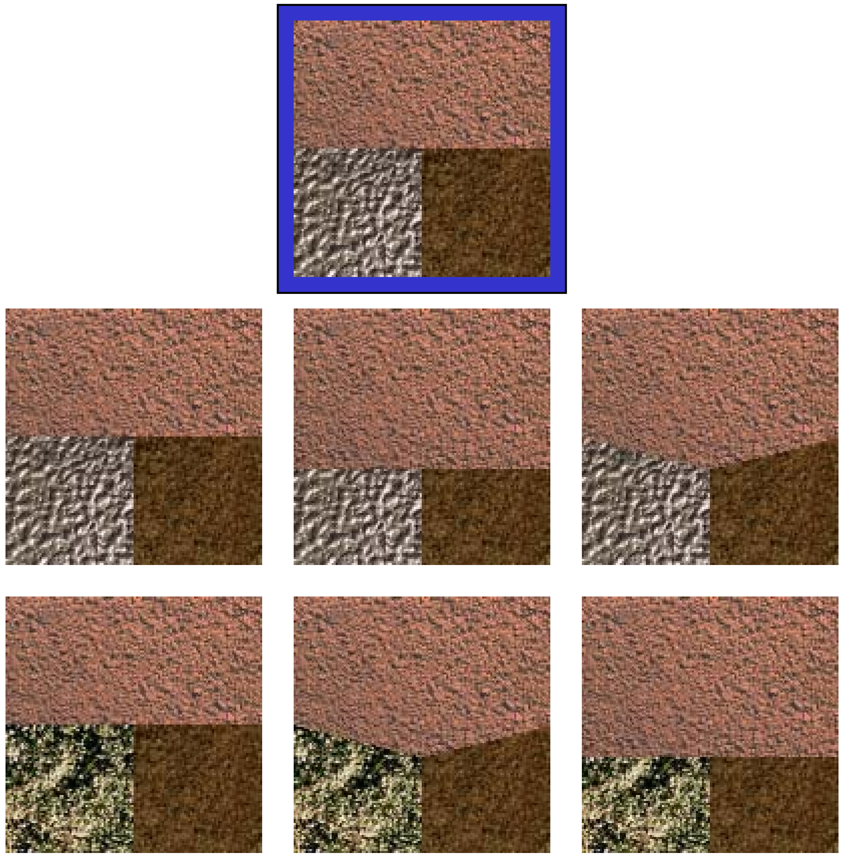


Figure 22. Retrieval Results for Natural Texture Mosaics Database I. The Image at the Top with Blue Border Is the Query Image. All Other Images Are Ranked in the Order of Similarity with the Query Image from Left to Right, Top to Bottom.

Table 3. S Values for Retrieval Results Shown in Figure 22.

Retrieved Image	**S Value**
Top Left	0.0000
Top Middle	2.2433
Top Right	2.8607
Bottom Left	4.3782
Bottom Middle	4.4079
Bottom Right	4.5265

8.2.2 Experiments using Natural Texture Mosaics Database II

In Figure 23 through Figure 32, examples of the performance for the Natural Texture Mosaics Database II are shown. In each figure, the image at the top left with the blue border is the query image. The five most similar images retrieved from the database are shown in the order of similarity from left to right and top to bottom. Note that all retrieved images have common texture types with the query image. Table 4 through Table 13, respectively, list the corresponding S values for the images.

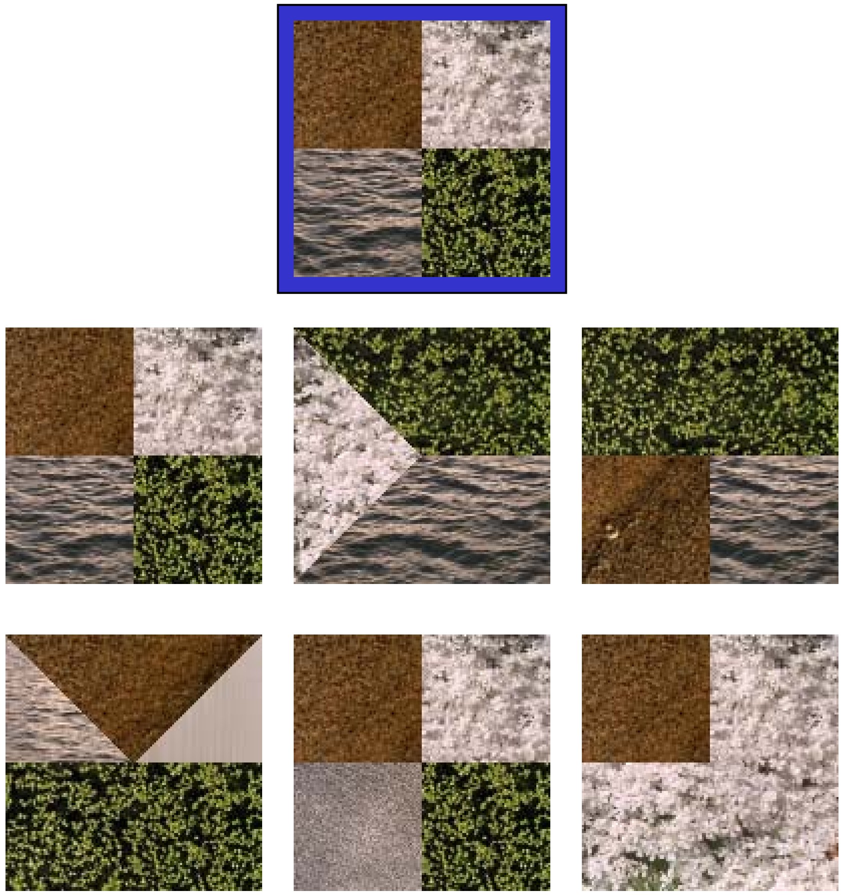


Figure 23. First Example of Retrieval Results for Query of the Natural Texture Mosaics Database II. The Image at the Top with Blue Border Is the Query Image. All Other Images Are Ranked in the Order of Similarity with the Query Image from Left to Right, Top to Bottom.

Table 4. S Values for Retrieval Results Shown in Figure 23.

Retrieved Image	S Value
Top Left	0.0000
Top Middle	3.3322
Top Right	3.8086
Bottom Left	5.7541
Bottom Middle	6.0538
Bottom Right	8.9020

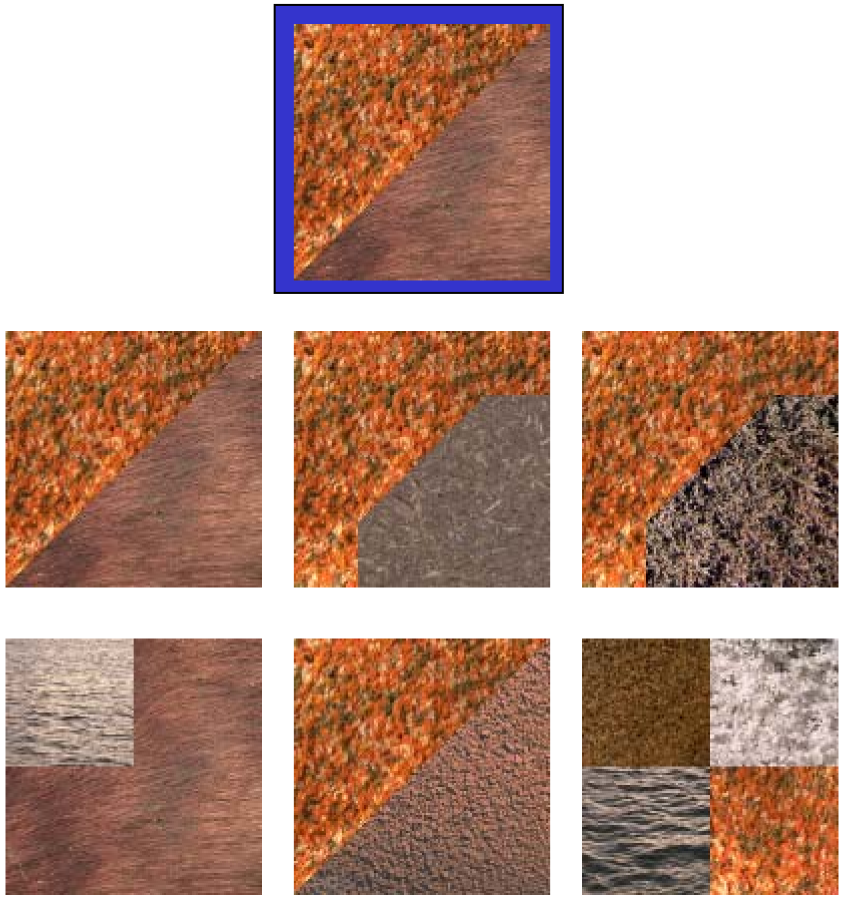


Figure 24. Second Example of Retrieval Results for Query of the Natural Texture Mosaics Database II.

Table 5. S Values for Retrieval Results Shown in Figure 24.

Retrieved Image	**S Value**
Top Left	0.0000
Top Middle	5.2009
Top Right	5.6328
Bottom Left	5.8508
Bottom Middle	7.8500
Bottom Right	16.4595

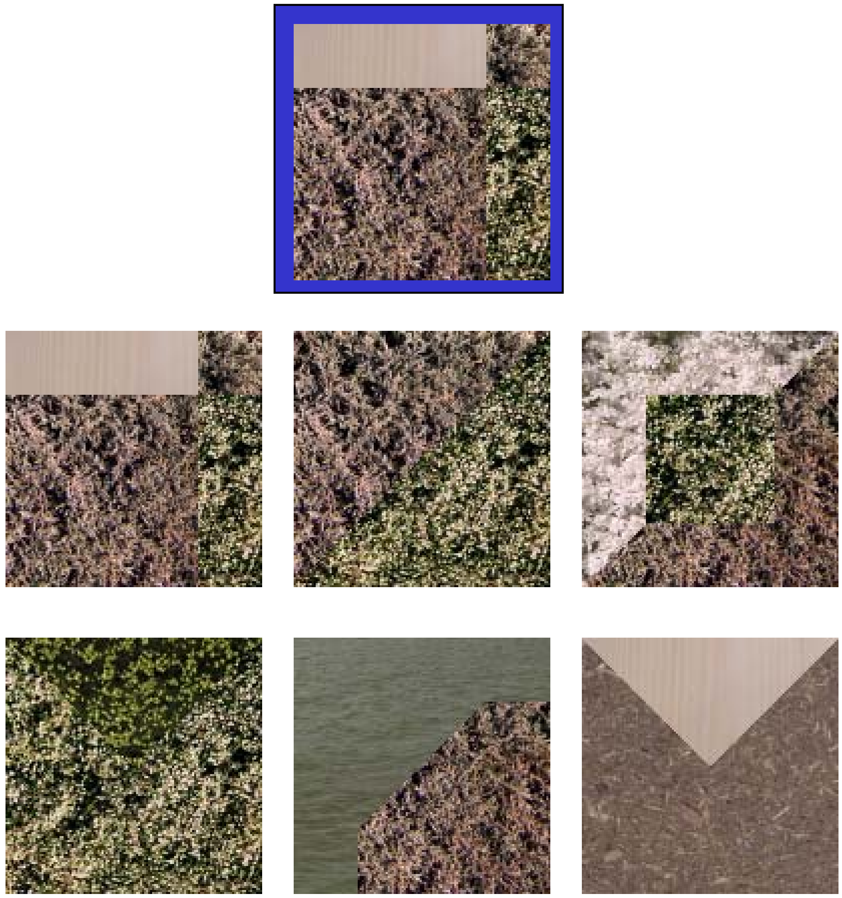


Figure 25. Third Example of Retrieval Results for Query of the Natural Texture Mosaics Database II.

Table 6. S Values for Retrieval Results Shown in Figure 25.

Retrieved Image	S Value
Top Left	0.0000
Top Middle	2.4232
Top Right	10.6481
Bottom Left	16.2232
Bottom Middle	18.5311
Bottom Right	19.0535

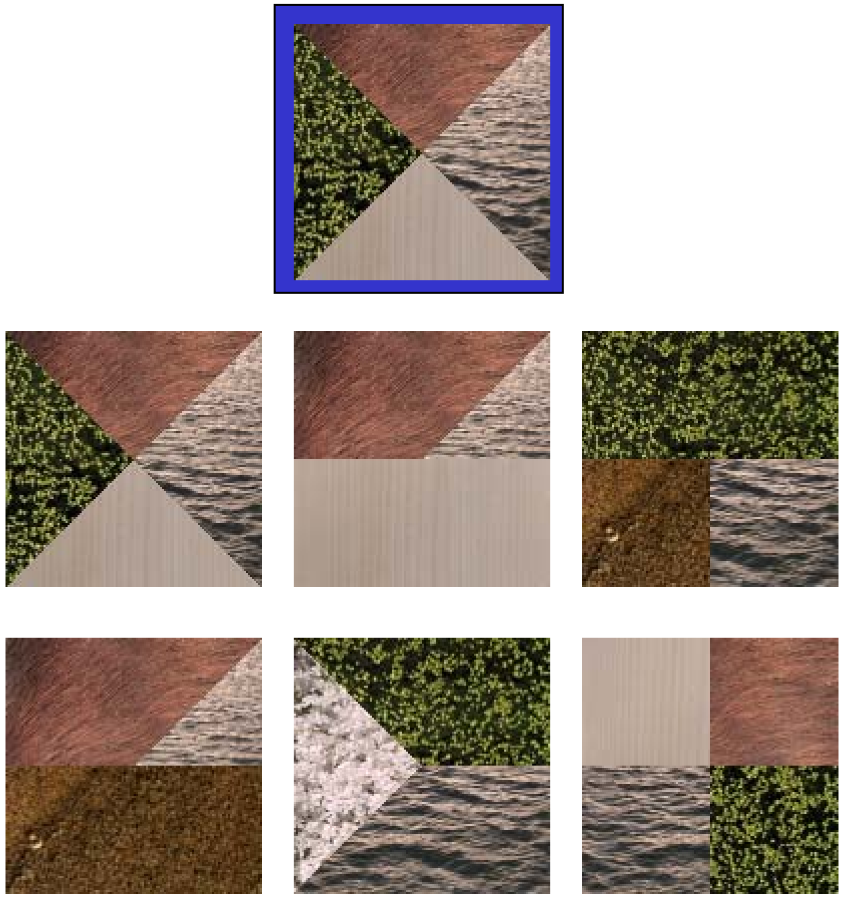


Figure 26. Fourth Example of Retrieval Results for Query of the Natural Texture Mosaics Database II.

Table 7. S Values for Retrieval Results Shown in Figure 26.

Retrieved Image	**S Value**
Top Left	0.0000
Top Middle	7.0583
Top Right	7.8229
Bottom Left	8.2077
Bottom Middle	8.4442
Bottom Right	10.0138

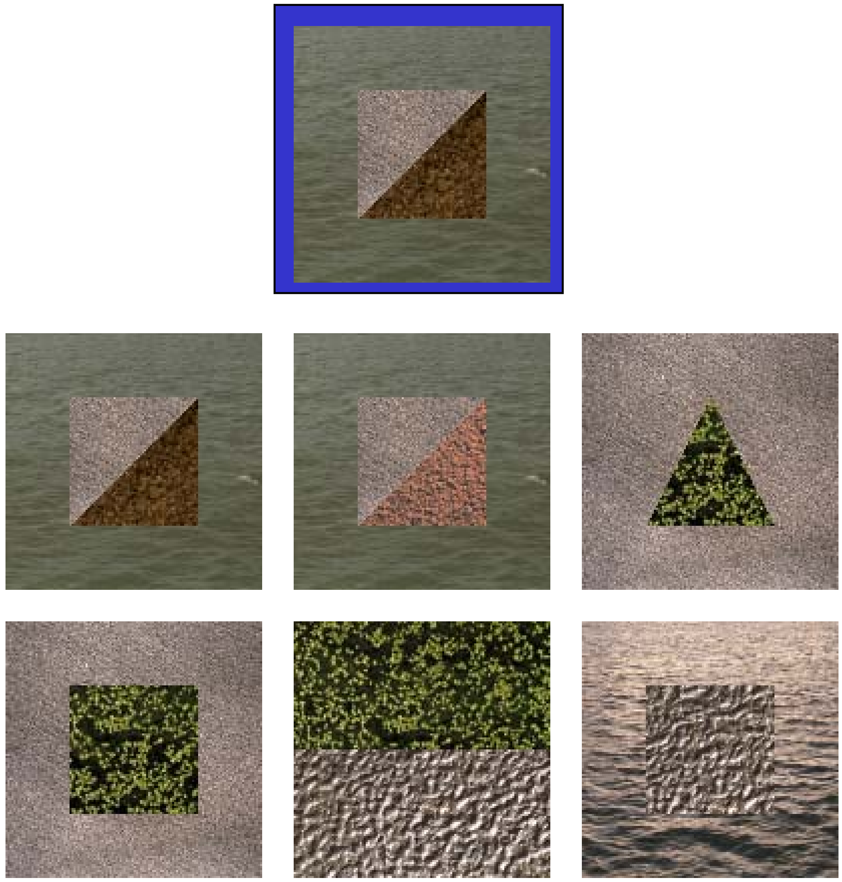


Figure 27. Fifth Example of Retrieval Results for Query of the Natural Texture Mosaics Database II.

Table 8. S Values for Retrieval Results Shown in Figure 27.

Retrieved Image	**S Value**
Top Left	0.0000
Top Middle	7.0109
Top Right	17.2405
Bottom Left	17.2579
Bottom Middle	17.3736
Bottom Right	19.3084

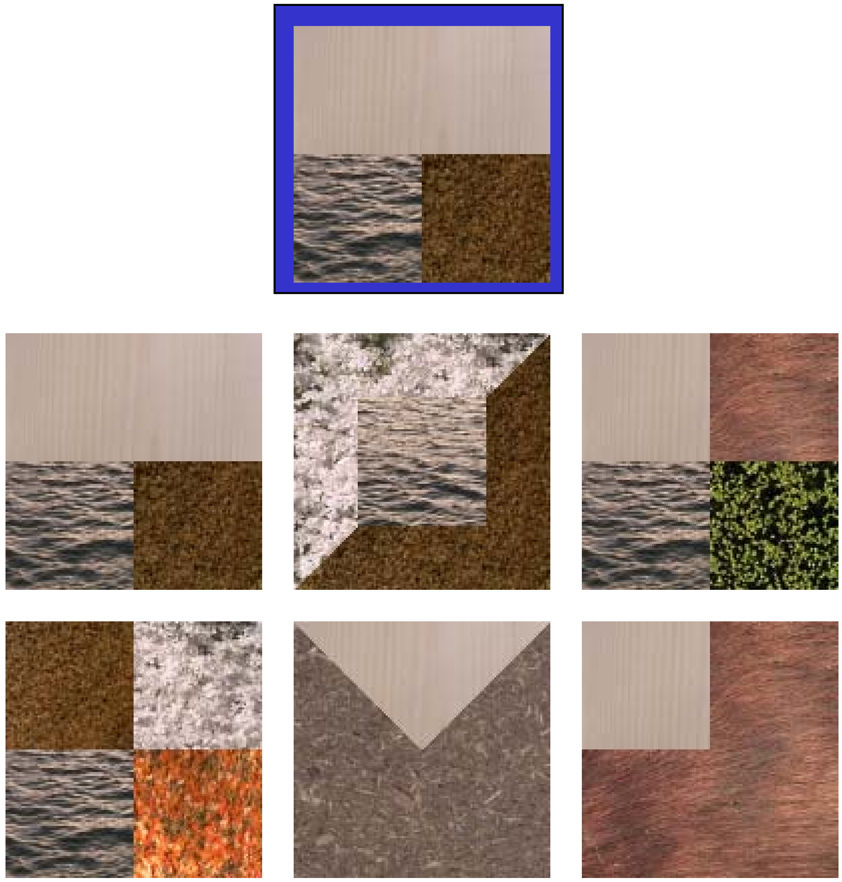


Figure 28. Sixth Example of Retrieval Results for Query of the Natural Texture Mosaics Database II.

Table 9. S Values for Retrieval Results Shown in Figure 28.

Retrieved Image	**S Value**
Top Left	0.0000
Top Middle	3.8043
Top Right	4.0063
Bottom Left	4.5774
Bottom Middle	17.5186
Bottom Right	20.3577

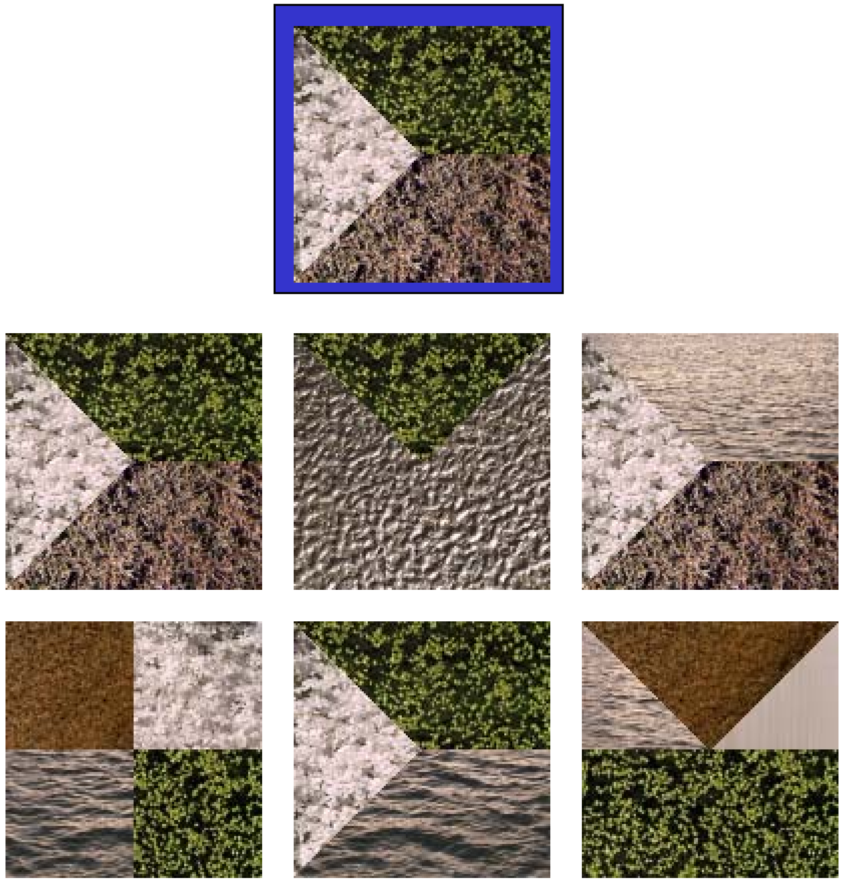


Figure 29. Seventh Example of Retrieval Results for Query of the Natural Texture Mosaics Database II.

Table 10. S Values for Retrieval Results Shown in Figure 29.

Retrieved Image	**S Value**
Top Left	0.0000
Top Middle	3.4077
Top Right	5.4130
Bottom Left	6.2624
Bottom Middle	6.3576
Bottom Right	6.6369

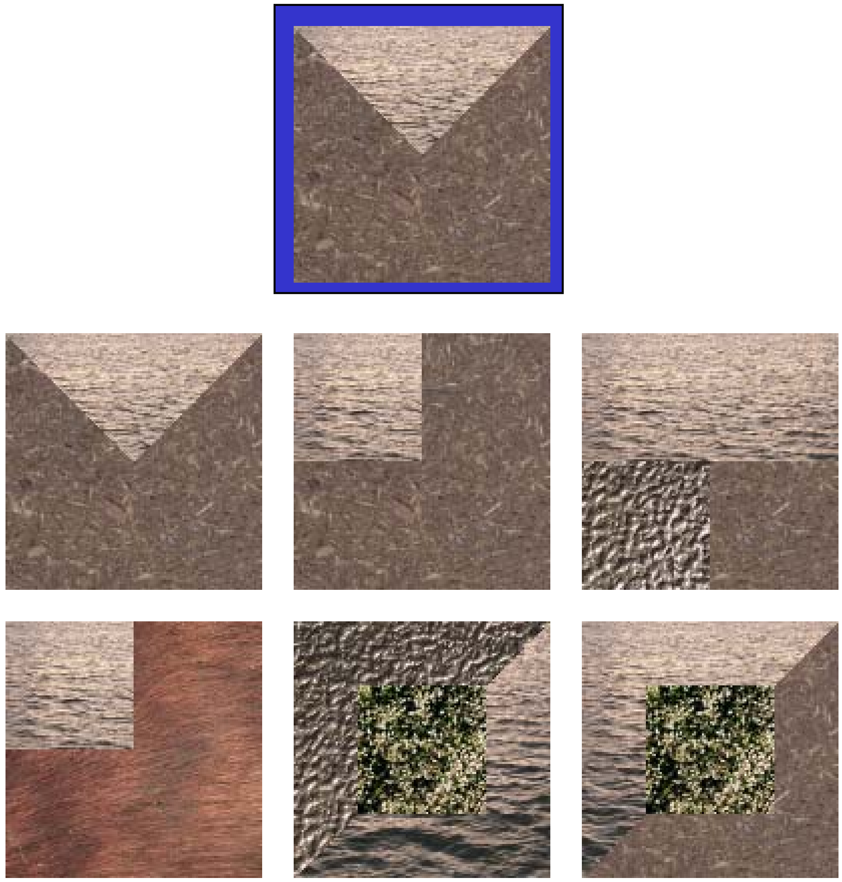


Figure 30. Eighth Example of Retrieval Results for Query of the Natural Texture Mosaics Database II.

Table 11. S Values for Retrieval Results Shown in Figure 30.

Retrieved Image	**S Value**
Top Left	0.0000
Top Middle	2.9550
Top Right	3.1231
Bottom Left	3.2208
Bottom Middle	8.8206
Bottom Right	9.0158

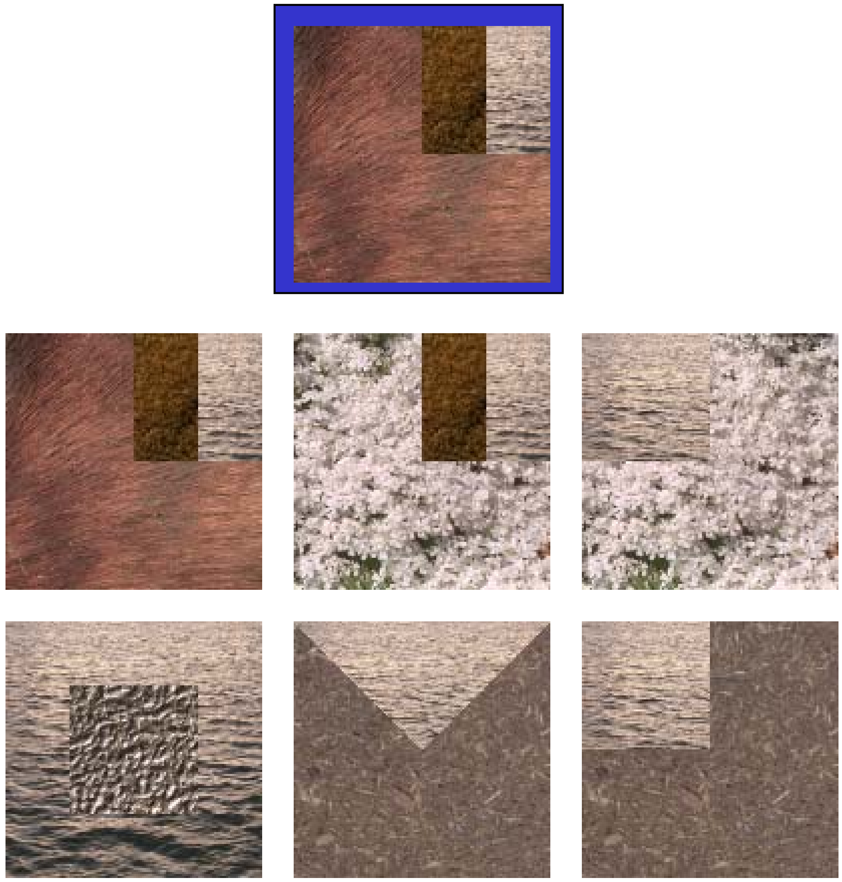


Figure 31. Ninth Example of Retrieval Results for Query of the Natural Texture Mosaics Database II.

Table 12. S Values for Retrieval Results Shown in Figure 31.

Retrieved Image	**S Value**
Top Left	0.0000
Top Middle	3.0497
Top Right	20.2907
Bottom Left	20.6235
Bottom Middle	21.0742
Bottom Right	21.1236

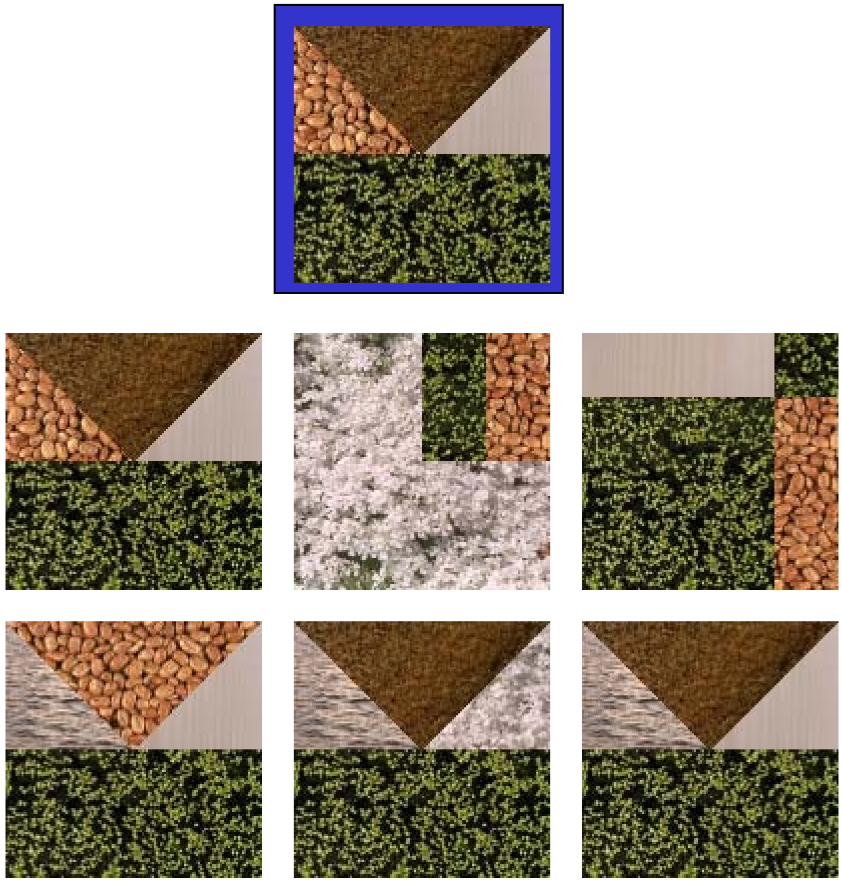


Figure 32. Tenth Example of Retrieval Results for Query of the Natural Texture Mosaics Database II.

Table 13. S Values for Retrieval Results Shown in Figure 32.

Retrieved Image	S Value
Top Left	0.0000
Top Middle	5.1033
Top Right	5.8570
Bottom Left	10.8644
Bottom Middle	17.4452
Bottom Right	64.3375

8.2.3 Experiments using the Natural Scenes Database

In Figure 33 through Figure 40, similar examples are shown for the Natural Scene Database images. Again note that the exact match of the query image is ranked as the closest retrieved image. The other retrieved images all contain similar scenery as the query image.

In each figure, the image with a blue border is the query image. The rest of the images in this figure are selected from the database by the algorithm for being the most similar images to the query image. They are also ranked from most similar to least similar from left to right and top to bottom. The image in the database that is exactly the same as the query image is retrieved and ranked as the most similar to the query image. Table 14 through Table 21, respectively, list the corresponding S values for the images.

It should also be noted that for the results presented in Figure 40, a subset of plants, vegetation, and flower images is extracted from the database. This is done, because this subset is distinctly different from the rest of the database. However for all the other results shown, the entire database is considered, and the plants, vegetation, and flower images are always effectively excluded from the retrieval results when the query image is not a plant, vegetation, or flower image.

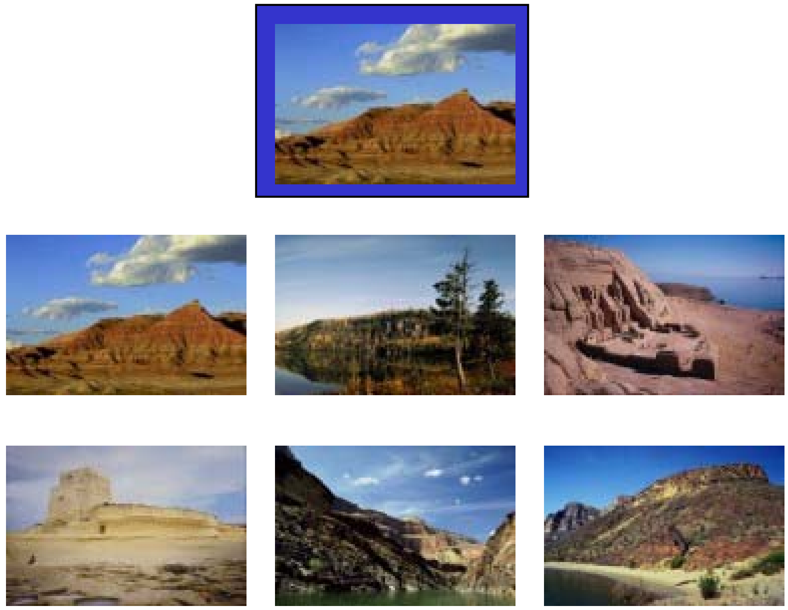


Figure 33. First Example of Retrieval Results for Query of the Natural Scenes Database. The Image at the Top Left with Blue Border Is the Query Image. The Other Images from Left to Right, Top to Bottom Are Retrieved Images with Decreasing Similarity.

Table 14. S Values for Retrieval Results Shown in Figure 33.

Retrieved Image	S Value
Top Left	0.0000
Top Middle	2.8690
Top Right	4.4552
Bottom Left	5.4820
Bottom Middle	6.1939
Bottom Right	6.8532

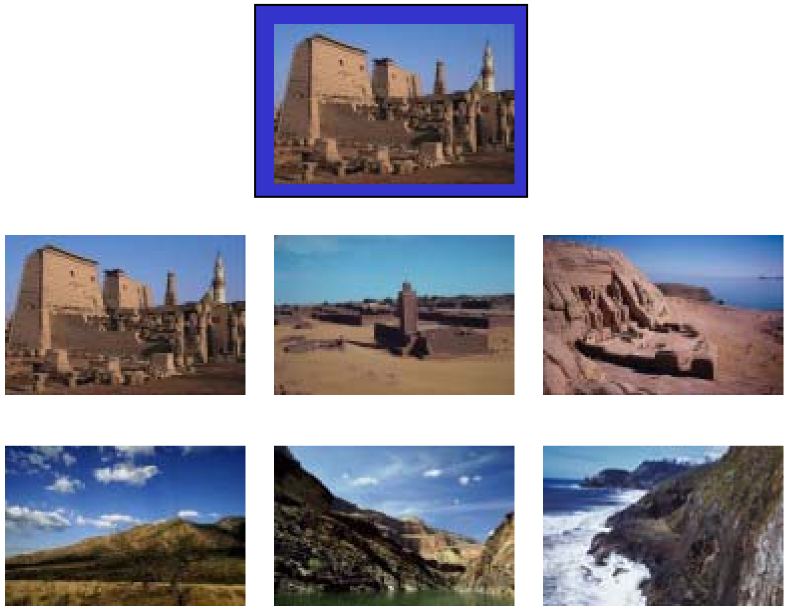


Figure 34. Second Example of Retrieval Results for Query of the Natural Scenes Database.

Table 15. S Values for Retrieval Results Shown in Figure 34.

Retrieved Image	**S Value**
Top Left	0.0000
Top Middle	13.2759
Top Right	13.4045
Bottom Left	13.4643
Bottom Middle	13.6548
Bottom Right	13.6583

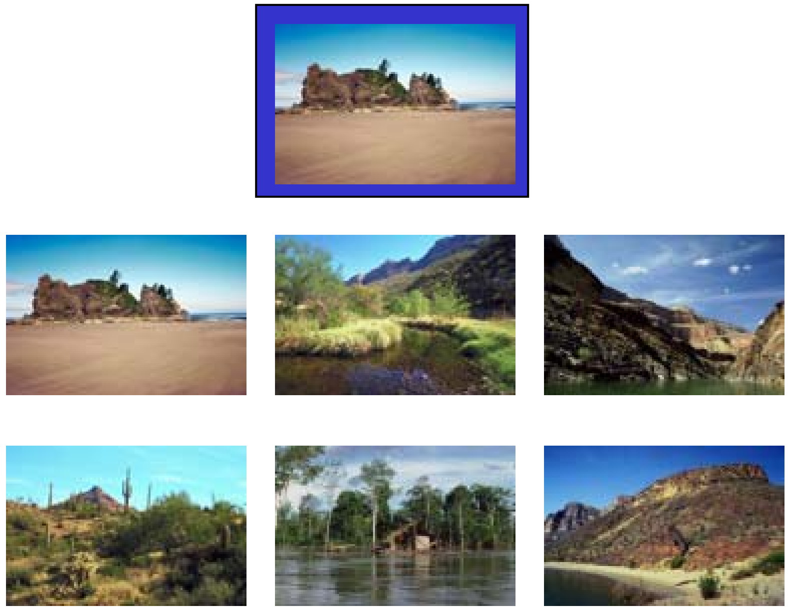


Figure 35. Third Example of Retrieval Results for Query of the Natural Scenes Database.

Table 16. S Values for Retrieval Results Shown in Figure 35.

Retrieved Image	**S Value**
Top Left	0.0000
Top Middle	3.5480
Top Right	3.8671
Bottom Left	4.0440
Bottom Middle	5.4373
Bottom Right	9.4634

Figure 36. Fourth Example of Retrieval Results for Query of the Natural Scenes Database.

Table 17. S Values for Retrieval Results Shown in Figure 36.

Retrieved Image	**S Value**
Top Left	0.0000
Top Middle	3.5950
Top Right	5.7966
Bottom Left	5.9901
Bottom Middle	6.4883
Bottom Right	6.5316

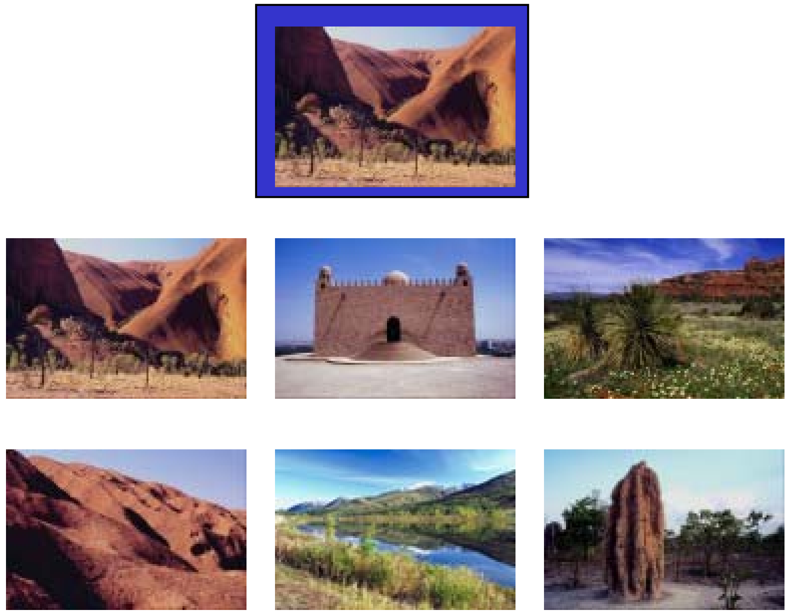


Figure 37. Fifth Example of Retrieval Results for Query of the Natural Scenes Database.

Table 18. S Values for Retrieval Results Shown in Figure 37.

Retrieved Image	**S Value**
Top Left	0.0000
Top Middle	5.1507
Top Right	5.7584
Bottom Left	8.1685
Bottom Middle	9.7732
Bottom Right	11.3665

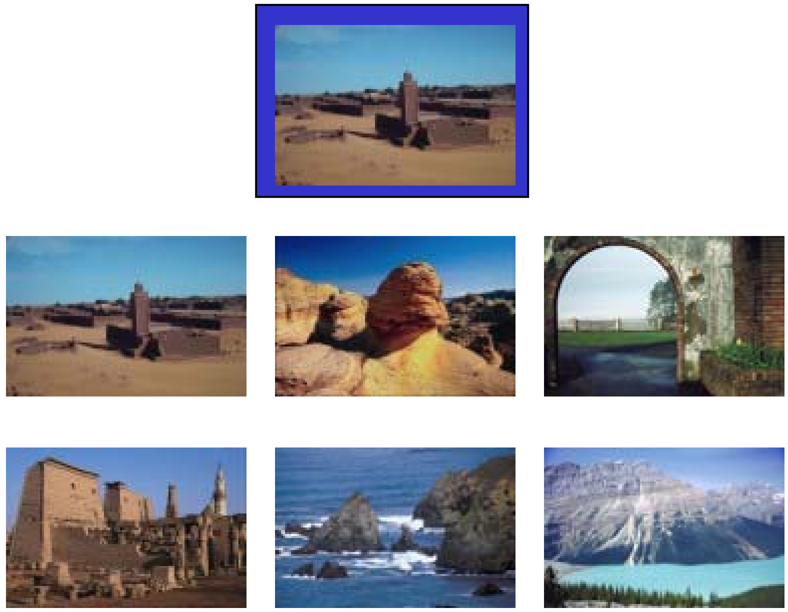


Figure 38. Sixth Example of Retrieval Results for Query of the Natural Scenes Database.

Table 19. S Values for Retrieval Results Shown in Figure 38.

Retrieved Image	**S Value**
Top Left	0.0000
Top Middle	9.8102
Top Right	10.0595
Bottom Left	11.1587
Bottom Middle	11.3357
Bottom Right	11.4376

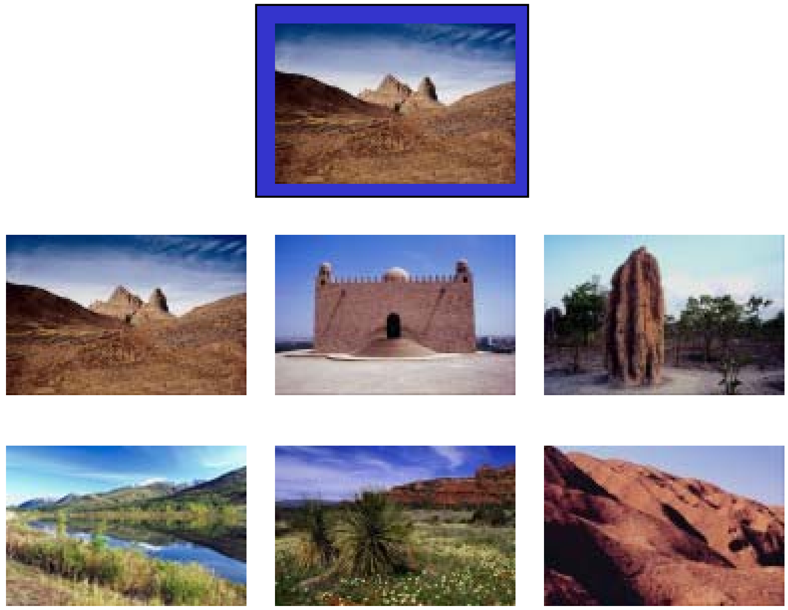


Figure 39. Seventh Example of Retrieval Results for Query of the Natural Scenes Database.

Table 20. S Values for Retrieval Results Shown in Figure 39.

Retrieved Image	**S Value**
Top Left	0.0000
Top Middle	15.7918
Top Right	16.5850
Bottom Left	17.9803
Bottom Middle	18.6936
Bottom Right	19.6895

Figure 40. Eighth Example of Retrieval Results for Query of the Natural Scenes Database.

Table 21. S Values for Retrieval Results Shown in Figure 40.

Retrieved Image	**S Value**
Top Left	0.0000
Top Middle	2.3322
Top Right	3.8578
Bottom Left	7.4531
Bottom Middle	7.7316
Bottom Right	7.9724

8.3 Performance Evaluation

It should be noted that the segmentation of database images can be done off-line. The online retrieval process involving the segmentation of the query image and computation and ranking of the corresponding S measures takes around 60 seconds on a 400 MHz Intel Pentium III class machine with 128 MBytes of random access memory (RAM).

A performance evaluation exercise is conducted for this system. This consists of evaluating the "wall clock" run time for the database characterization and the query processes. This run time data is collected for the three different test databases, and on two different hardware compute platforms. The results are presented in Table 22. The following clarification points should be kept in mind, as the data in Table 22 is examined:

- In the process (**PROC.**) column, OFF-LINE refers to the off-line process of characterizing the database, and QUERY refers to the on-line process of retrieving similar images to the query image from the database.
- In the database (**DB**) column, TEXT. I, TEXT. II, and SCENES refer to the "Natural Texture Mosaics Database I", a sample of the "Natural Texture Mosaics Database II", and a sample of the "Natural Scenes Database," respectively.
- The **IMAGES** column describes how many images are in the database.
- The **SIZE** column describes the size of the images in the database in pixels (length by height). All the images in a database are of the same size, and of course, different databases can have the images of different sizes.

- The **AREA** column describes the area of the images in the database in pixels.
- The hardware (**HW**) column describes the platform that is used to run the evaluations. Platform "A" refers to a 233 MHz Intel Pentium class machine with 48 Mbytes of random access memory (RAM). Platform "B" refers to a 400 MHz Intel Pentium III class machine with 128 Mbytes of RAM.
- The **TIME** column describes the "wall clock" time it takes to run the process: Characterization or Query. This is in minutes.
- The **PER IMAGE** column is the normalized "wall clock" run time to one image. It is the "per image in the database" run time. This is in minutes.

Table 22. Performance Analysis Results.

PROC.	DB	IMAGES	SIZE	AREA	HW	TIME	PER IMAGE
OFF-LINE	TEXT. I	30	128 X 128	16,384	A	376	12.53
OFF-LINE	TEXT. I	30	128 X 128	16,384	B	143	4.77
OFF-LINE	TEXT. II	102	128 X 128	16,384	A	1,269	12.44
OFF-LINE	TEXT. II	102	128 X 128	16,384	B	465	4.56
OFF-	SCENES	51	120 X 80	9,600	A	357	7.00

LINE							
OFF-LINE	SCENES	51	120 X 80	9,600	B	118	2.31
QUERY	TEXT. I	30	128 X 128	16,384	A	12	0.40
QUERY	TEXT. I	30	128 X 128	16,384	B	3	0.10
QUERY	TEXT. II	102	128 X 128	16,384	A	15	0.15
QUERY	TEXT. II	102	128 X 128	16,384	B	4	0.04
QUERY	SCENES	51	120 X 80	9,600	A	6	0.12
QUERY	SCENES	51	120 X 80	9,600	B	2	0.04

After studying the data in the table above, the following observations can be made:

- For a specific compute platform, the time it takes to characterize a database is linear with the number of images in the database.
- The time that it takes to characterize an image is linear with the area of the image.
- The time it takes to perform a query does not depend on the size of the database, but on the size and area of the query image.
- Both the characterization of multiple images in a database, and the extraction of C^3T features for the windows across the image being segmented for characterization, are independent processes. Therefore, this system lends itself

to being implemented on parallel processing platforms, where the execution speedup achieved will be linear with the number of parallel processors employed.

8.4 Conclusions

In this chapter, experimental results demonstrating the efficacy of the proposed retrieval algorithm are presented and discussed. The experiments are carried out using the three databases described in Chapter 3. Performance evaluations are also conducted for different stages of the process in different compute platforms.

Chapter 9

SUMMARY AND CONCLUSIONS

This work focuses on retrieval of color images that are similar to an inquiry image. The similarity is based on the color texture content and geometrical attributes of different textural regions in the images. The Multispectral Simultaneous Autoregressive (MSAR) random field model with a 4-neighbor set is used to characterize the textures. An additional feature set consisting of ratios of sample means of the true color planes is also used to represent the color content. The combination of these two feature sets, which in this case is a 22-dimensional feature vector, is used throughout this work to represent each color texture region. This feature set is referred to as C^3T (Color Content Color Texture).

An unsupervised color texture segmentation algorithm is developed in conjunction with the C^3T features. These features are extracted from a sampling window that slides over the entire image. The extracted features are clustered in the 22-dimensional space using a histogram-based clustering algorithm that automatically detects the number of significant clusters. By mapping back the detected clusters to the spatial domain of an image, distinct regions of homogeneous color texture are identified and segmented out. The developed segmentation algorithm is a novel aspect of this work, and through experimentation with a number of synthetic as well as real images, its efficacy is established.

The other new aspect of this work involves the development of a new retrieval process. This process utilizes the described segmentation algorithm and the distinct textural regions that are identified. As the first step, the database images that have similar textures to those of the query image are identified. This is done by applying the same clustering algorithm to the C^3T features that are extracted from the MFS (Maximum Fitting Square) to each segmented region of the query image and database images. Note that the process of extracting the MFS C^3T features for database images is done off-line. Once the subsets of images having one or more common textures with the query image are identified, the rest of the database images are removed from further consideration resulting in a "Reduced Search Space".

A new similarity measure is established to quantify the closeness of two images. This measure utilizes texture as well as spatial size and arrangement of the image regions. The centroid and parameters of the best fitting ellipse to a region are used to represent the spatial characteristics of each region.

Through a region association approach, that involves C^3T features, each region of the query image is associated with one of the regions of a database image. Once this matching is completed, an overall similarity measure (S) involving both textural and spatial characteristics is computed. By ranking the S measures, those images that are most similar to the query image can be identified and retrieved.

The performance of the approach is tested on three separate databases, two of which contain mosaics of natural textures and the third one consisting of real natural scenes. In each case, the query image is included in the database to be searched and it always is found as the most similar image. The visual inspection of the results do confirm the effectiveness of the developed retrieval system, and demonstrate its capability in identifying visually close images to the query image.

It should be noted that the entire retrieval process is unsupervised, and the only required input is the query image.

9.1 Directions for Future Research

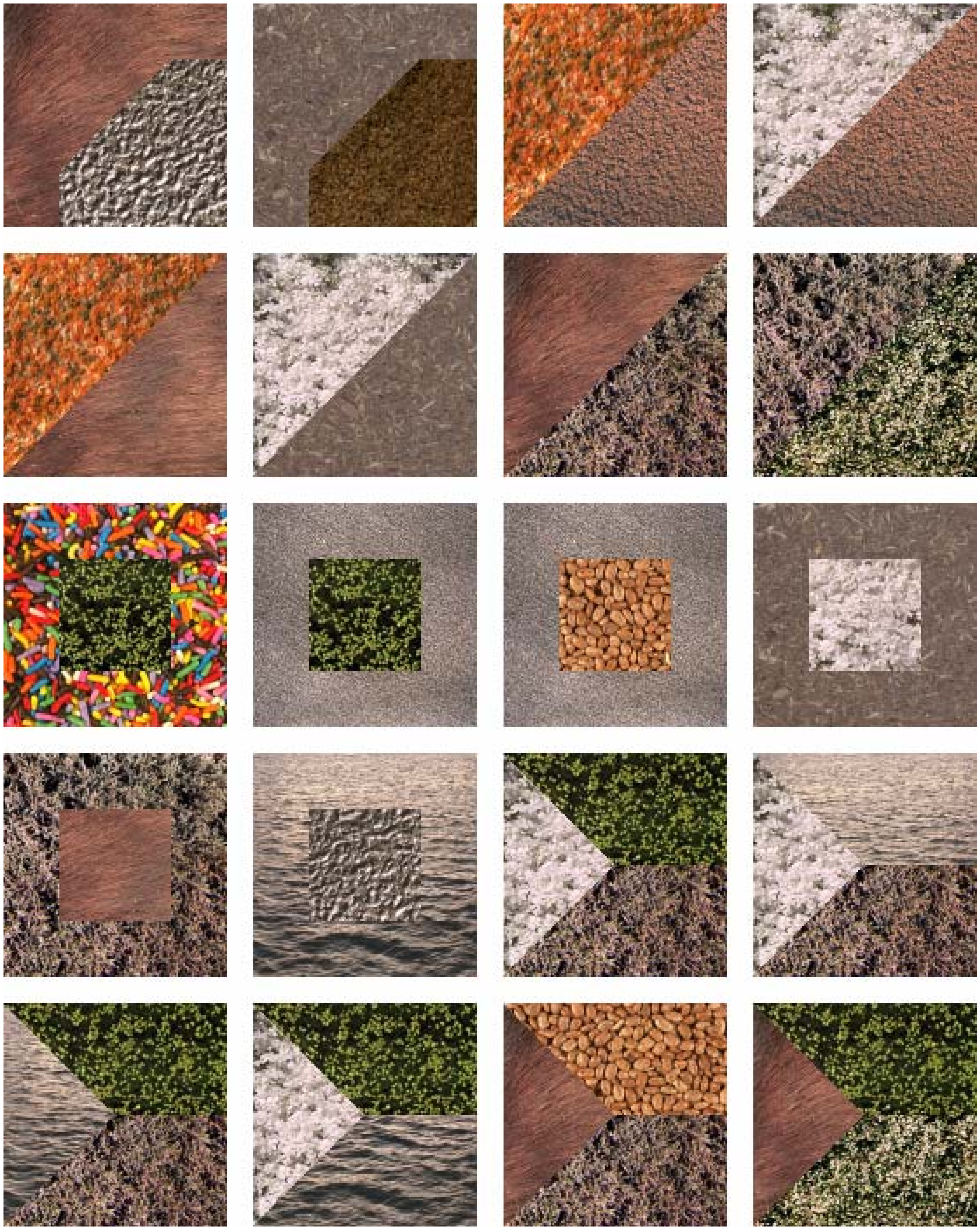
It is hoped that this body of work forms the basis for future research. Potential candidate topics for this are the following:

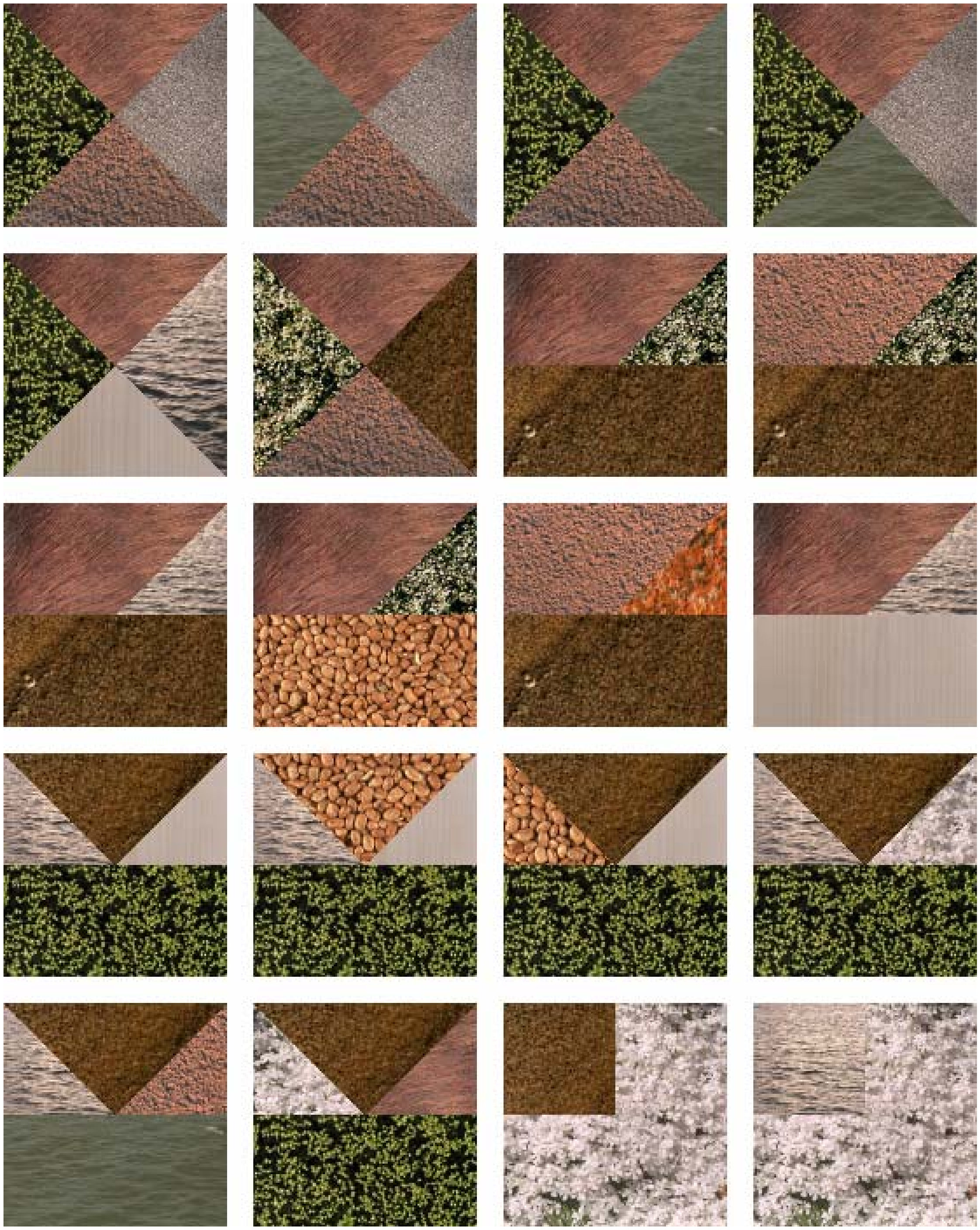
- This method uses texture and color features for image segmentation. There is no usage of object based features. And additional step can be to add object-based features to image segmentation approach.
- The proposed system uses query by example. An extension of it can use query by description as well.
- Application of the segmentation method to video event detection processes.
- Extending the application of texture and color based segmentation to video processing, in conjunction with object-based features and/or motion based features.
- Video database manipulation systems, like video retrieval systems, based on this framework.

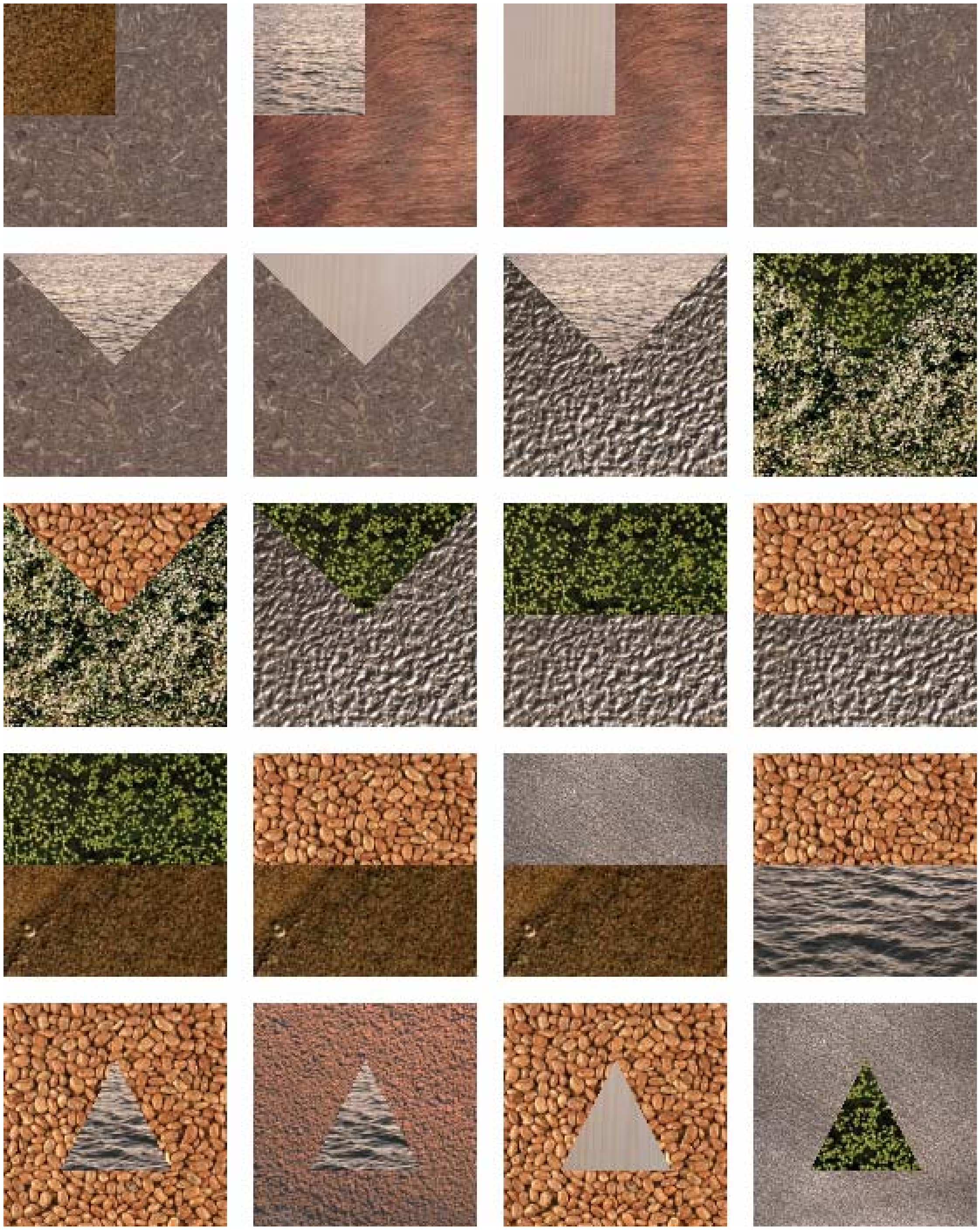
APPENDIX

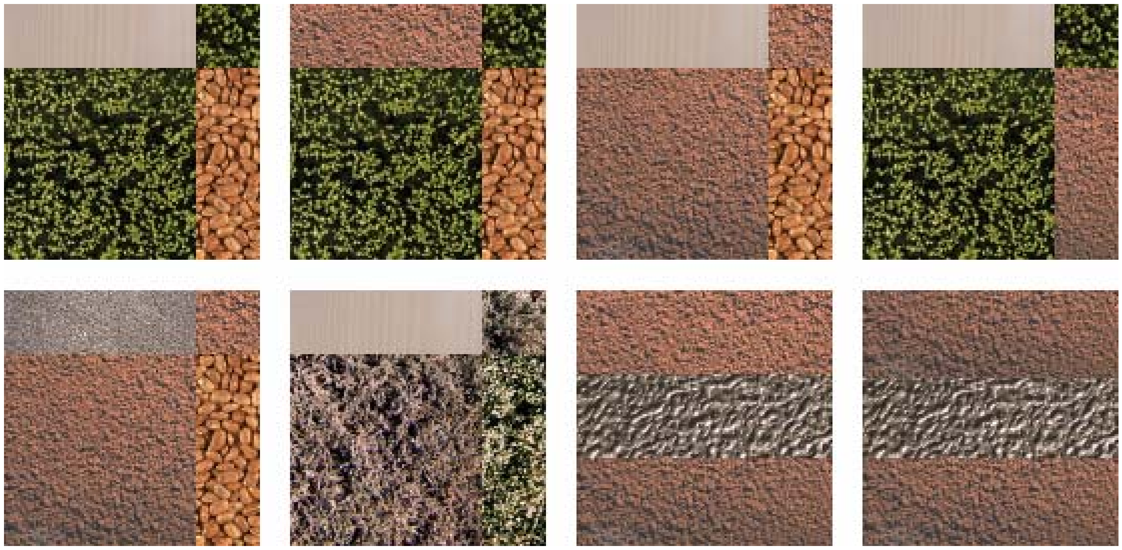
APPENDIX A

NATURAL TEXTURE MOSAICS DATABASE II



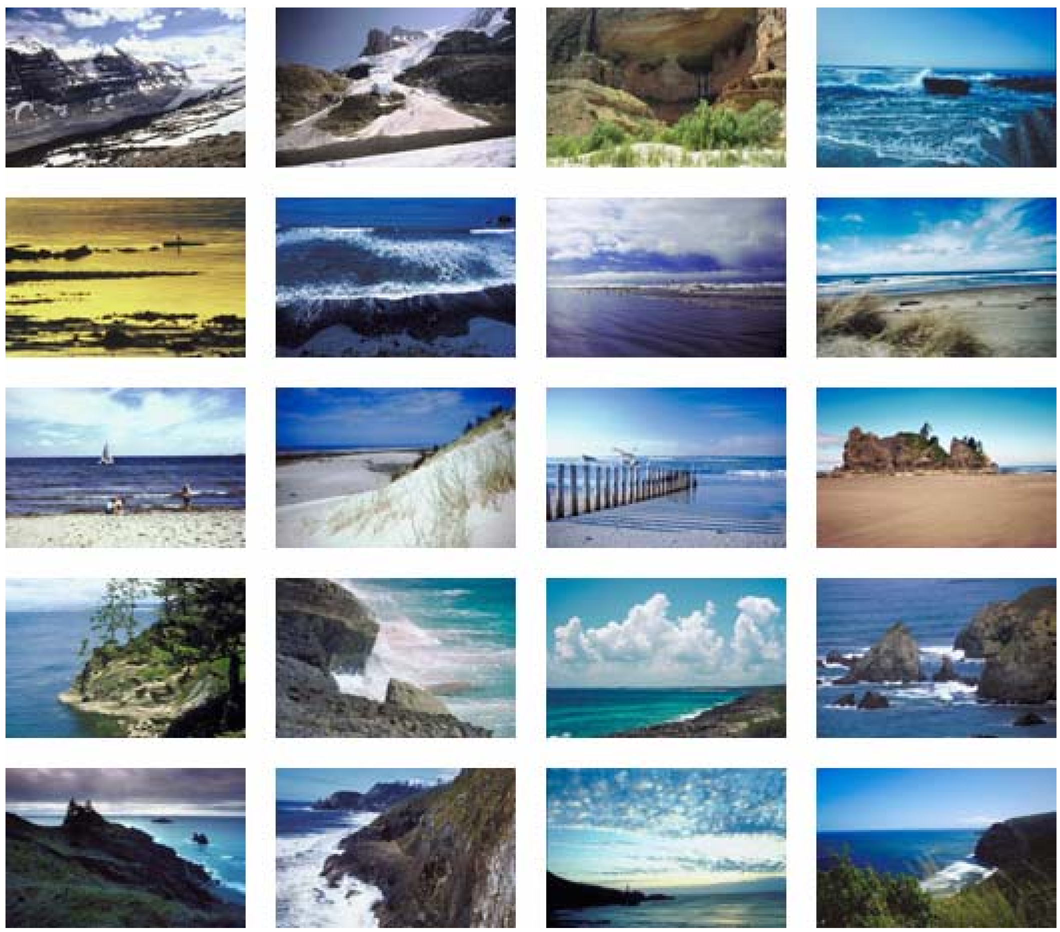


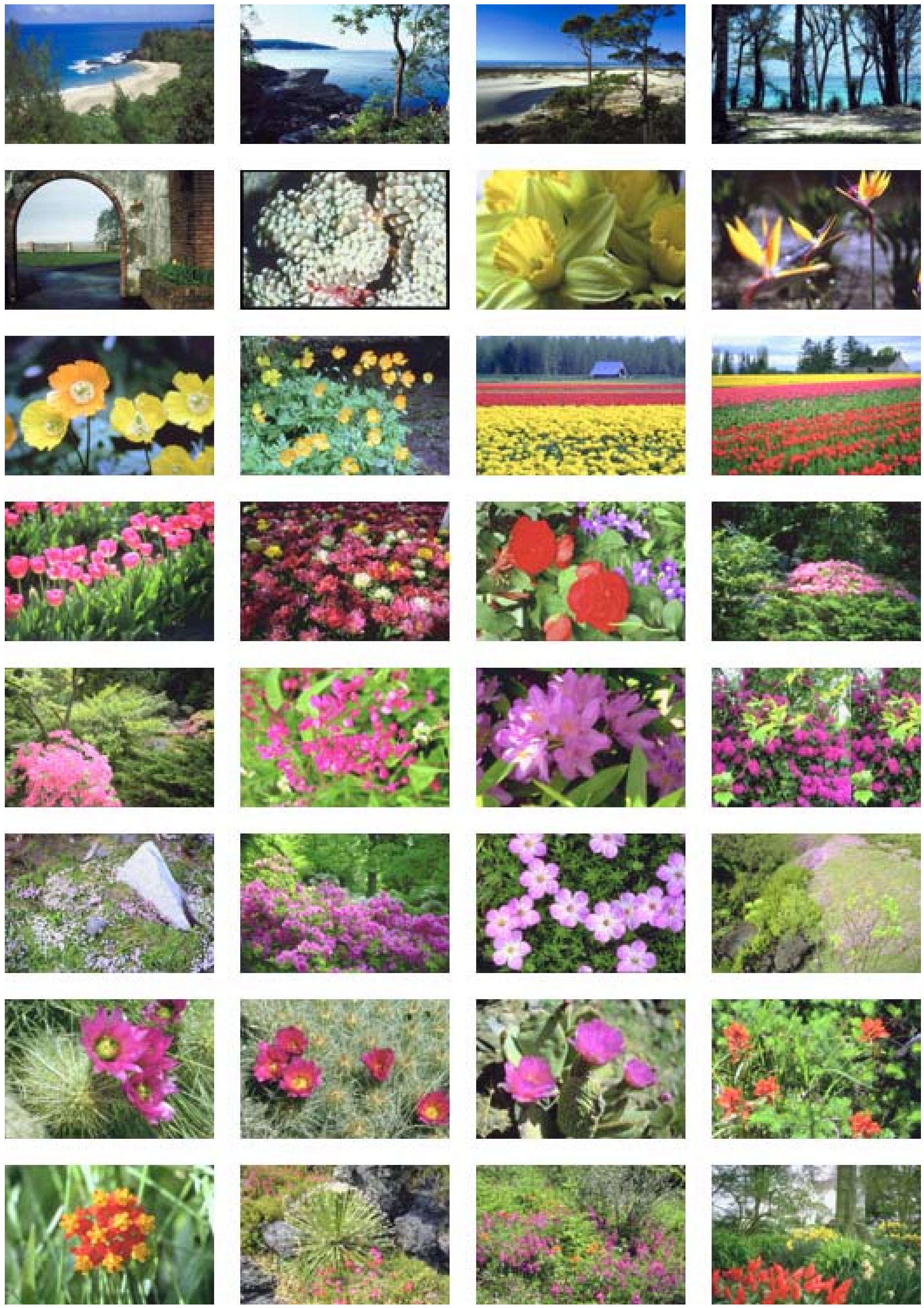


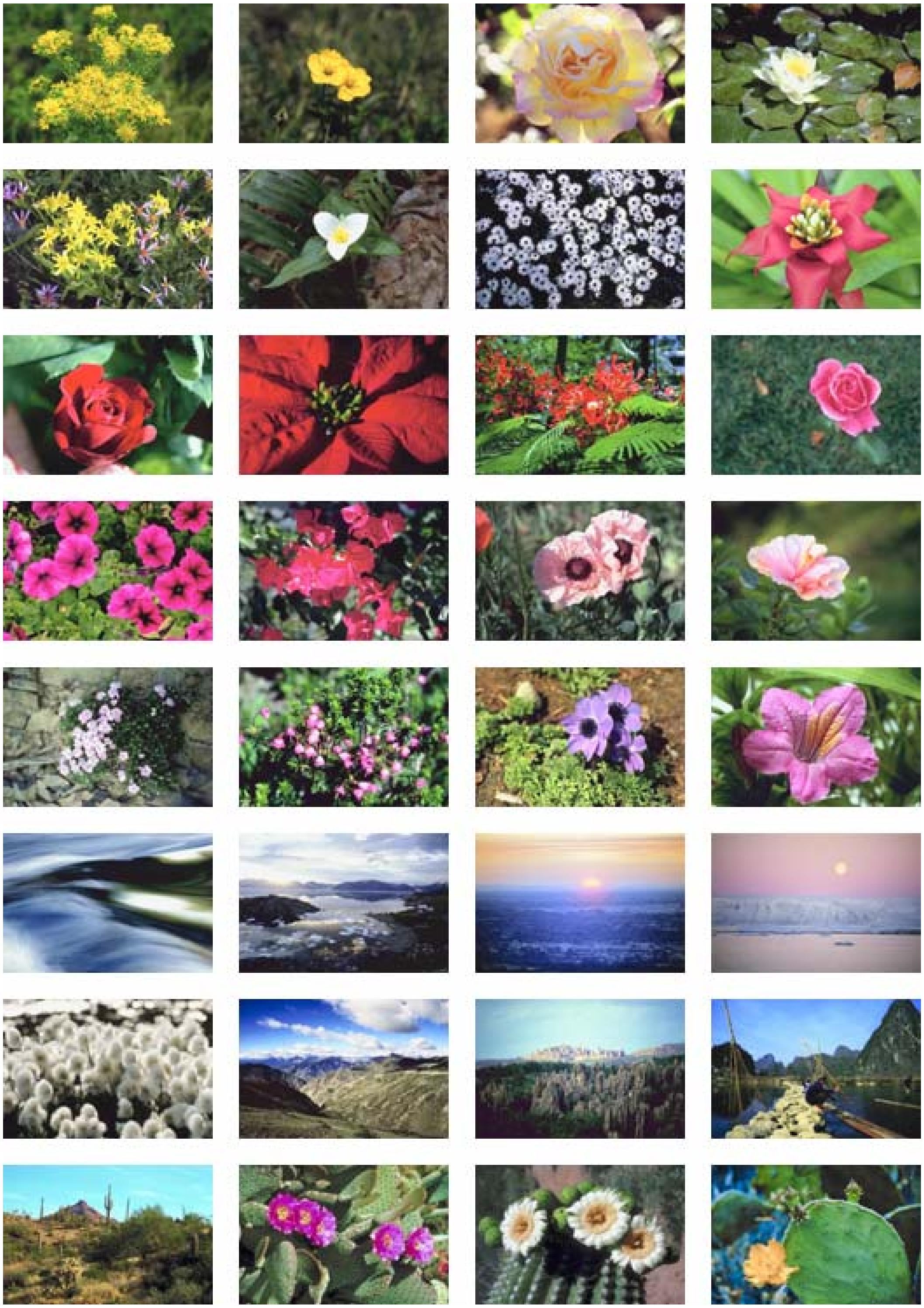


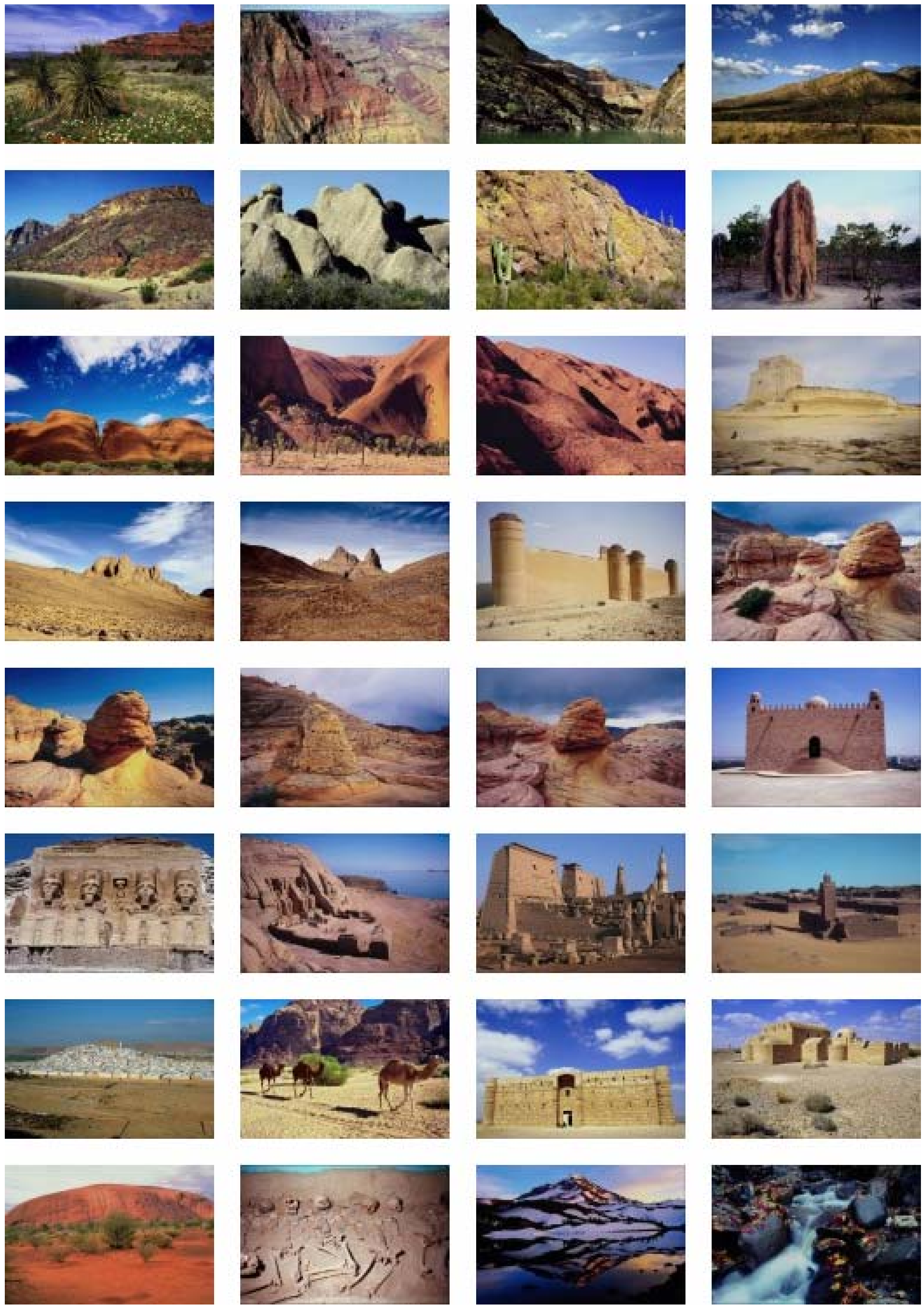
APPENDIX B

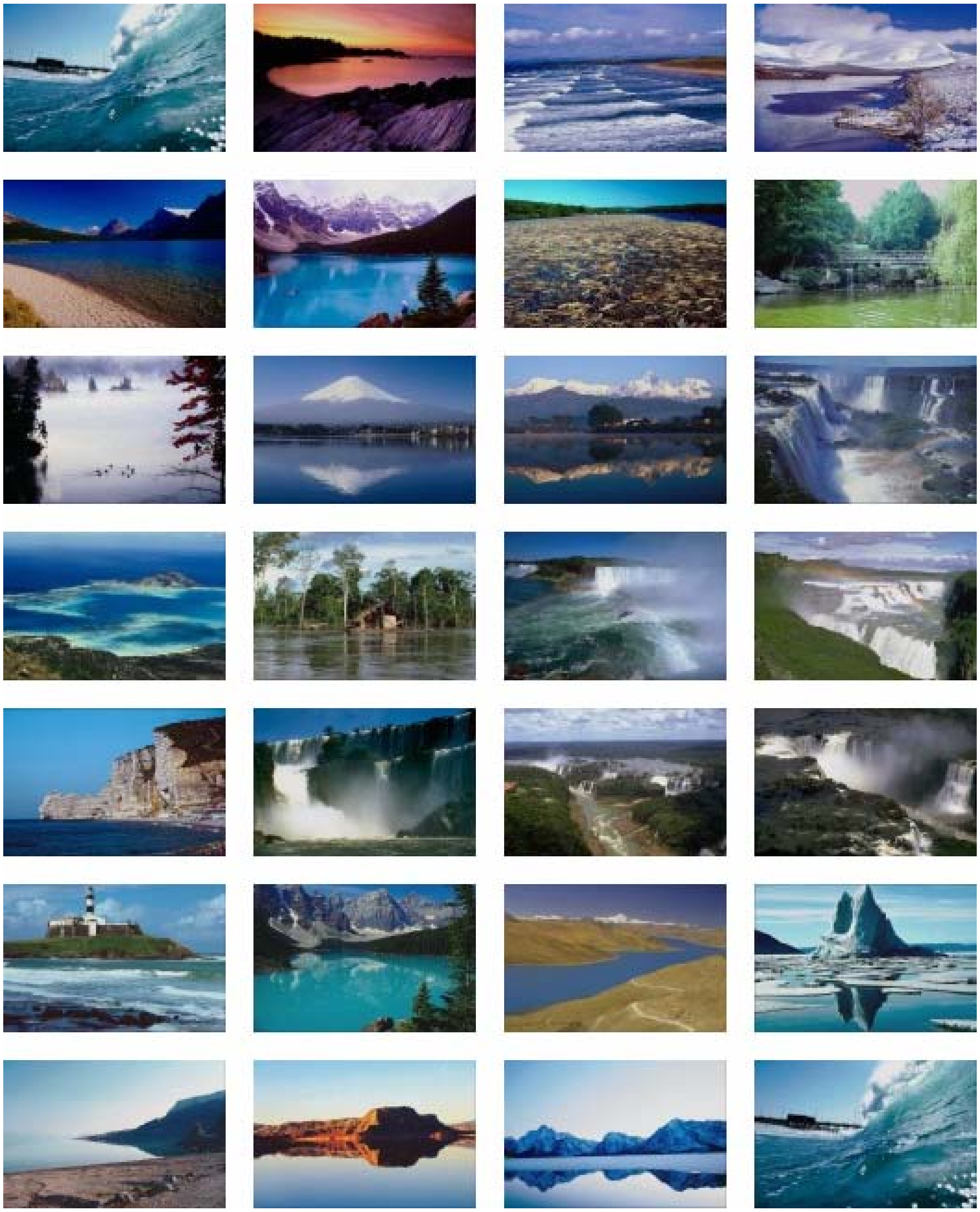
NATURAL SCENES DATABASE

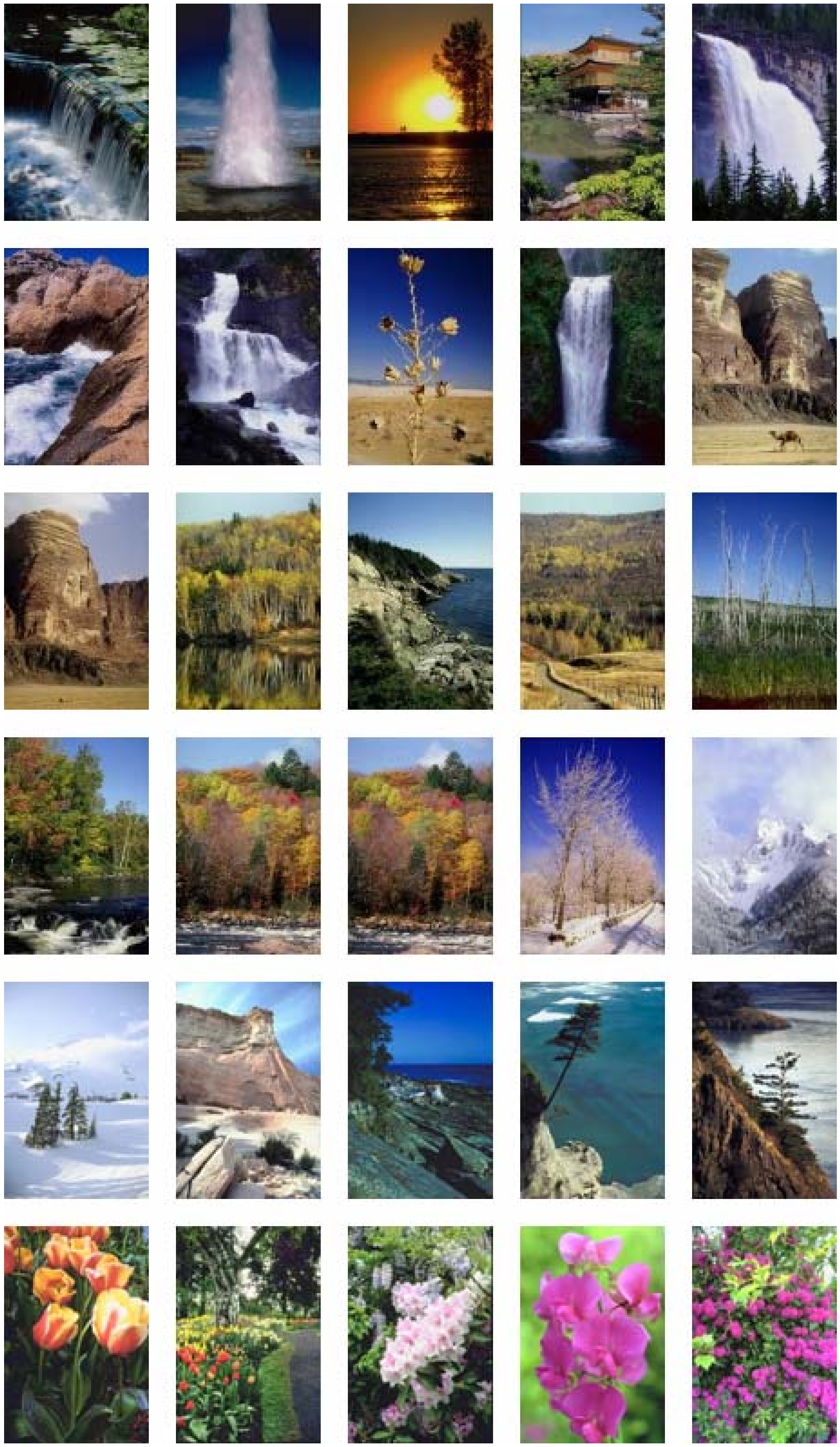


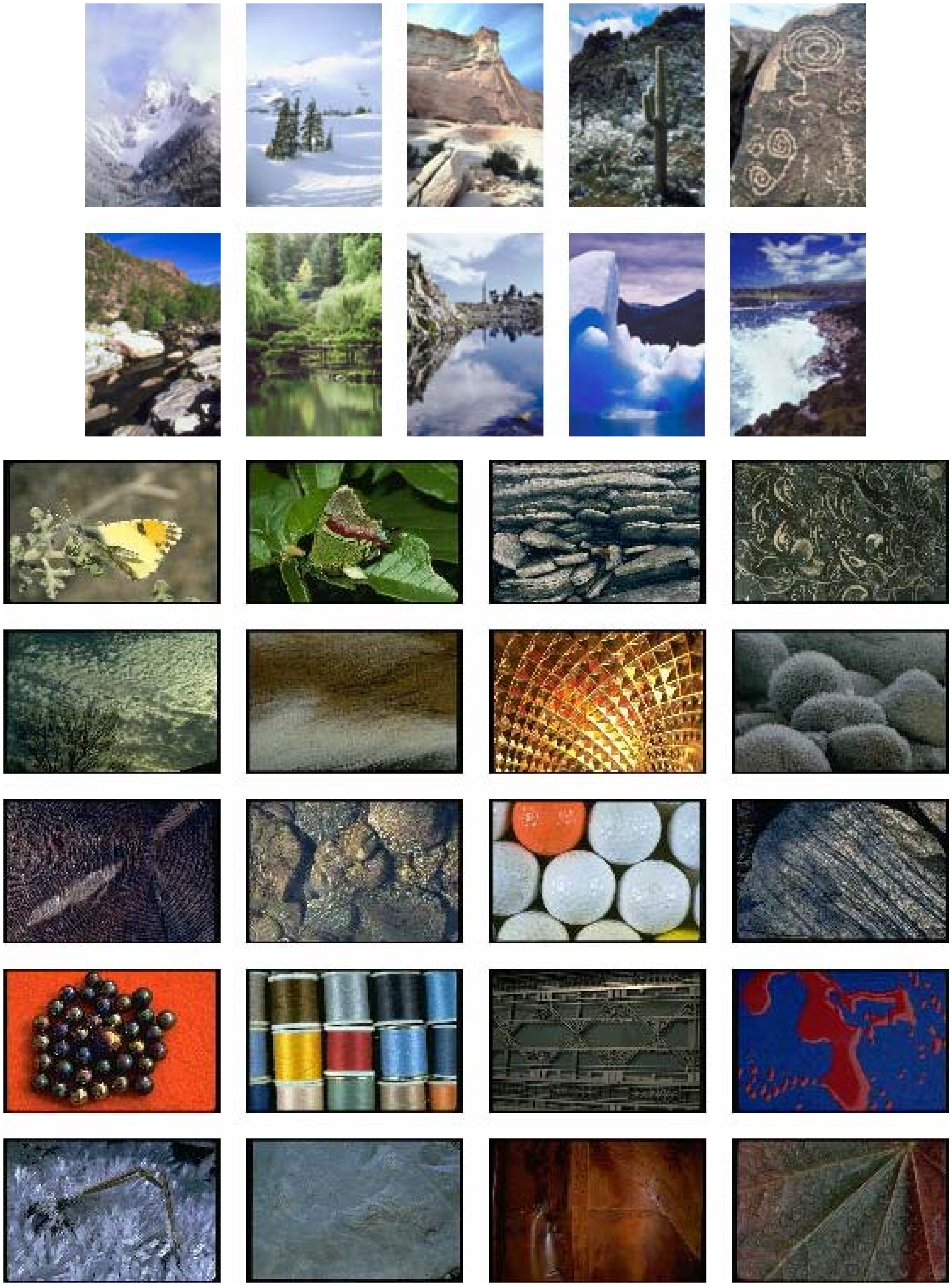


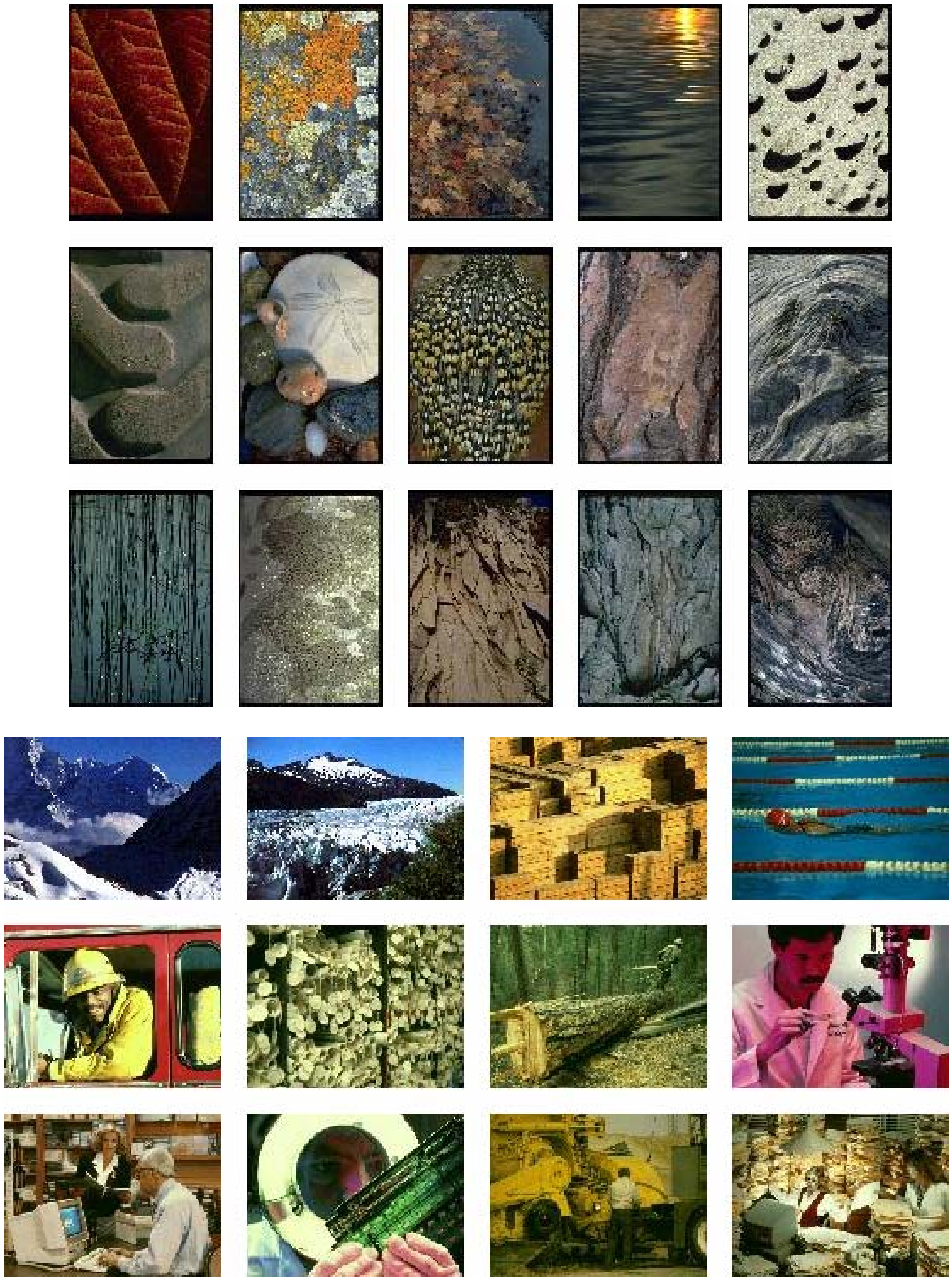












REFERENCES

[1] D. L. Swets and J. Weng, "Using Discriminant Eigenfeatures for Image Retrieval," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, August 1996, pp. 831-836.

[2] M. Shneier and M. Abdel-Mottaleb, "Exploiting the JPEG Compression Scheme for Image Retrieval," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, August 1996, pp. 849-853.

[3] G. Healey and A. Jain, "Retrieving Multispectral Satellite Images Using Physics-Based Invariant Representations," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, August 1996, pp. 842-848.

[4] S. A. Chien and H. B. Mortensen, "Automating Image Processing for Scientific Data Analysis of a Large Image Database," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, August 1996, pp. 854-859.

[5] B. S. Manjunath and W. Y. Ma, "Texture Features for Browsing and Retrieval of Image Data," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, August 1996, pp. 837-842.

[6] Y. Rui and T. S. Huang, "Image Retrieval; Current Techniques, Promising Directions, and Open Issues," *Journal of Visual Communications and Image Representation*, vol. 10, 1999, pp. 39-62.

[7] J. W. Bennett and A. Khotanzad, "Multispectral Random Field Models for Synthesis and Analysis of Color Images," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 20, no. 3, March 1998, pp. 327-332.

[8] Y. Deng and B. S. Manjunath, "Unsupervised Segmentation of Color-Texture Regions in Images and Video," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 23, no. 8, August 2001, pp. 800-810.

[9] J. R. Smith and S-F. Chang, "Integrated Spatial and Feature Image Query," *Multimedia Systems – ACM - © Springer-Verlag 1999*, vol. 7, no. 2, March 1999, pp. 129-140.

[10] K. B. Eom, "Segmentation of Monochrome and Color Textures Using Moving Average Modeling Approach," *Elsevier Science B. V. Image and Vision Computing*, no. 17, 1999, pp. 233-244.

[11] M. Mirmehdi and M. Petrou, "Segmentation of Color Textures," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, no. 2, February 2000, pp. 142-159.

[12] A. Mojsilović, J. Hu, and R. J. Safranek, "Perceptually Based Color Texture Features and Metrics for Image Retrieval," *Proceedings of the International Conference on Image Processing*, vol. 3, 1999, pp. 588-592.

[13] A. Mojsilović, J. Kovačević, J. Hu, R. J. Safranek, and S. K. Ganapathy, "Matching and Retrieval Based on the Vocabulary and Grammar of Color Patterns," *IEEE Trans. on Image Processing*, vol. 9, no. 1, January 2000, pp. 38-54.

[14] E. Saber and A. M. Tekalp, "Integration of Color, Shape, and Texture for Automatic Image Annotation and Retrieval," *Electronic Imaging (special issue)*, vol. 7, no. 3, July 1998, pp. 684-700.

[15] C. S. Fuh, S. W. Cho, and K. Essig, "Hierarchical Color Image Region Segmentation for Content-Based Image Retrieval System," *IEEE Trans. on Image Processing*, vol. 9, no. 1, January 2000, pp. 156-162.

[16] S. Santini and R. Jain, "Similarity Measures," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 21, no. 9, September 1999, pp. 871-883.

[17] A. Tversky, "Features of Similarity," *Psychological Review*, vol. 84, no. 4, July 1977, pp. 327-352.

[18] D. A. Adjeroh and M. C. Lee, "An Occupancy Model for Image Retrieval and Similarity Evaluation," *IEEE Trans. on Image Processing*, vol. 9, no. 1, January 2000, pp. 120-131.

[19] J. Li, J. Z. Wang, and G. Wiederhold, "IRM: Integrated Region Matching for Image Retrieval," *Proceedings of the ACM Multimedia 2000 Conference*, August 2000, pp. 147-156.

[20] S. Z. Li, K. L. Chan, and C. Wang, "Performance Evaluation of the Nearest Feature Line Method in Image Classification and Retrieval," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, November 2000, pp. 1335-1339.

[21] C. C. Yang, and M. C. Chan, "Color Image Retrieval Based on Textural and Chromatic Features," *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, vol. 4, 1999, pp. 922-927.

[22] International Business Machines (IBM) Company, "QBIC™ -- IBM's Query By Image Content," http://wwwqbic.almaden.ibm.com/.

[23] Vision and Modeling Group, MIT Media Laboratory, "Vision Texture (VisTex) database," http://www-white.media.mit.edu/vismod/, 1995.

[24] Corel Corporation, "Professional Photos CD-ROM Sampler – SERIES 200000," 1994.

[25] J. W. Bennett, "Modeling and Analysis of Gray Tone, Color, and Multispectral Texture Images by Random Field Models and Their Generalizations," Ph.D. Dissertation, *Southern Methodist University*, May 1997.

[26] A. Khotanzad and J. Y. Chen, "Unsupervised Segmentation of Textured Images by Edge Detection in Multidimensional Features," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 11, no. 4, April 1989, pp. 414-421.

[27] J. Y. Chen, "Segmentation of Textured Images by Edge Detection in Multidimensional Features," Ph.D. Dissertation, *Southern Methodist University*, September 1987.

[28] A. Khotanzad and A. Bouarfa, "Image Segmentation by a Parallel, Non-Parametric Histogram Based Clustering Algorithm," *Pattern Recognition*, vol. 23, no. 9, September 1990, pp. 961-963.

[29] R. C. Gonzalez and R. E. Woods, "Digital Image Processing," *Addison Wesley*, Book, 1992.

[30] A. Khotanzad and O. J. Hernandez, Sr., "Color Image Retrieval Using Multispectral Random Field Texture Models," *IEEE Digital Signal Processing Workshop*, October 15-18 2000, IEEE Signal Processing Society.

[31] A. Khotanzad and O. J. Hernandez, Sr., "Color Image Retrieval Using Multispectral Random Field Texture Model & Color Content Features," Paper submitted to *Pattern Recognition*, February 2002.