

Content Based Annotation and Retrieval in RAIDER

Stephanie Fountain

Department of Computer Science, University of Reading
Reading, RG6 6AY, England

Tieniu Tan

National Laboratory of Pattern Recognition,
Institute of Automation, Beijing, China

Abstract

A new system, RAIDER (Retrieval and Annotation of Image Databases), has been developed for the management of image databases. RAIDER was designed to combat the inadequacies and inefficiencies of traditional systems via a combination of content based retrieval and enhanced text based query capabilities. The interactive annotation technique employed in RAIDER is both quick and easy to use. As a whole RAIDER provides a flexible and efficient way to build and search image databases.

A system overview is given in this paper together with details of the rotation invariant texture analysis techniques developed for use in its implementation. Two methods of texture analysis are presented; a multichannel filtering technique based on Gabor filtering and an edge attribute method which utilises the Sobel edge operator. Retrieval and classification experiments are performed on a database of 1320 images taken from 44 Brodatz classes. The two methods resistance to Gaussian noise are characterised via content based retrieval experiments based on similar image queries. Finally an object selection tool (used during annotation) based on texture and colour analysis is presented. Experimental results are given throughout the paper where applicable.

1 Introduction

Image databases are becoming more pervasive with the advent of digital cameras and inexpensive storage media. The traditional method of retrieving images from a database is based on keyword searches, where the images are associated with a list of keywords describing their content which are referenced during a query. The annotation is subjective and the word list can never be comprehensive enough to cover every conceivable search pattern. If a search string does not appear in the list of keywords associated with an image then the image is not retrieved.

One solution to this problem is content based retrieval (CBR) in which the image content is analysed during a search thus avoiding the problems inherent in keyword searching. Colour, texture and shape measures can form the basis of a wide variety of query styles including those based on similar image properties to an example image. Intelligent image databases based on CBR have become a major research focus. General systems under development include QBIC [1] and Photobook [2]. More specialised examples are I²C [10], a system for indexing, storing and retrieving medical images, and MARCO [11], a system for retrieving maps by image content. The problem is also extended to video databases where millions of images are stored for each film. Systems such as JACOB [12] are currently under development to browse and query such databases.

The system discussed in this paper is RAIDER: Retrieval and Annotation of Image Databases. A combination of content based retrieval and enhanced text based query capabilities provides a unique solution to the database indexing problem. Images are efficiently added to the database via an interactive content based annotation process which reduces retrieval time and enhances RAIDER's knowledge of world objects. In specialised databases (e.g. textile designs) the annotation process can become fully automatic.

The implementation of RAIDER is still in its infancy. This paper gives an overview of the ideas behind RAIDER and presents the work completed to date. Content based retrieval can be regarded as a classification problem and annotation a classification and segmentation problem. Texture, colour and shape analysis can be used to solve the two problems. Texture analysis was selected as the main research area for this work as it is particularly suited to the analysis of outdoor scenes which are the preferred type of images for RAIDER.

The work focuses on an important but hitherto overlooked problem in image databases and texture analysis - the rotation invariant annotation and retrieval of texture images [6]. Two methods of accomplishing rotation invariance are presented: a multichannel filtering approach and an edge operator based method. Both are incorporated into RAIDER. The inclusion of rotation invariance distinguishes RAIDER from all other image

database systems. It also makes the system more consistent with human visual annotation and retrieval of images (as image recognition by the human visual system is clearly rotation invariant).

The remainder of the paper is organised as follows. The retrieval and annotation parts of the RAIDER system are presented. The rotation invariant texture analysis methods used in RAIDER are explained along with classification experiments and appropriate results. The methods are incorporated in to RAIDER for image retrieval and their resistance to Gaussian noise studied. Finally the most appropriate method is applied to object selection along with colour analysis for use in image annotation.

2 An Overview of RAIDER

A combination of content based retrieval techniques and text based queries is required in an ideal system. Content based queries are more flexible and not as subjective as those based on text. They do however take a longer period of time to execute as each image must be analysed in detail. Text based queries are required for speed (keyword matches are faster) and to enable searches for objects by name.

2.1 Retrieval

Figure 1 shows the current version of the retrieval section of RAIDER which accommodates both text and content based queries (which can be mixed and matched as required). Text based queries include object searches, (e.g. “Find me a picture of a house”) and image descriptions (e.g. “Find me a picture of a bright sunny day”). Content based queries can be based on colour, texture, shape, detail areas and similar image properties as explained below:

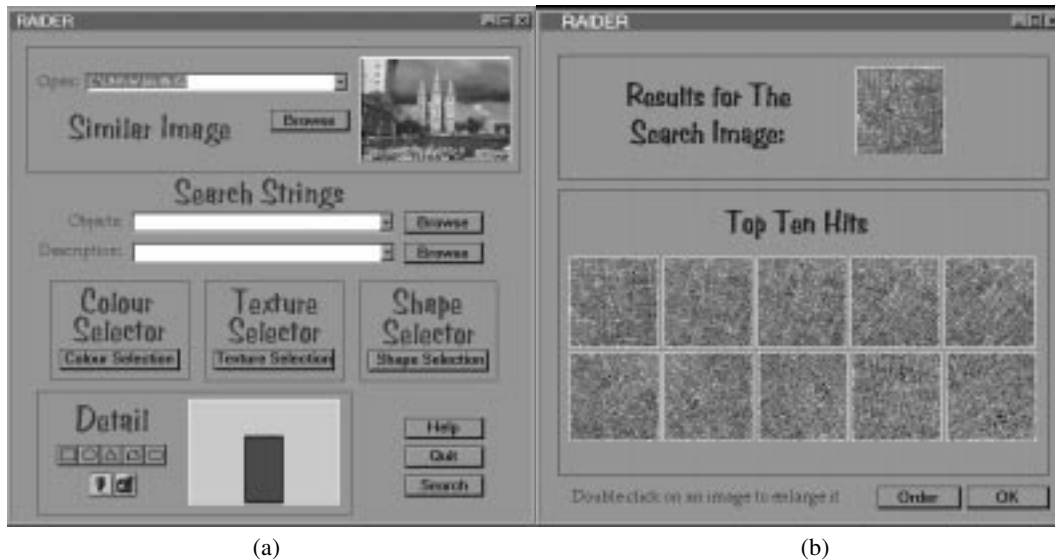


Figure 1: Content Based Image Retrieval in RAIDER

Colour: Either single or multiple search colours can be specified via the use of a colour wheel or predefined colour samples.

Texture: A texture selection tool enables the user to search for textures from a system library (i.e. textures RAIDER possesses knowledge of). User defined textures can be added to the library via the use of a filename entry box.

Shape: A drawing area is available for specifying shapes on which to search.

Detail area: The detail area drawing tool may be used in conjunction with the colour, texture and shape tools to specify areas in which the properties should occur. It can also be used separately to define required areas of dense image detail.

Similar Image Properties: Figure 1b shows the result of a ‘similar image’ search where a query image (shown at the top of the window) is presented to the system. Image features are computed and compared to those from all the images in the database. The n closest matches are selected and the corresponding images displayed as thumbnails in the lower half of the window. This method of retrieving images is the main focus of current work. The features used to retrieve the images are derived via texture analysis. The techniques used are described in Section 3.

2.2 Annotation

In order for text based queries to succeed the system must be familiar with the requested objects and keywords. Two extreme methods of accommodating such searches are: a) to annotate all images in the database and match keywords on querying (traditional method), and b) to maintain a descriptive list (feature vectors) of all known objects and use image analysis techniques to locate instances of the objects on querying.

Whilst method b is less time consuming at the data entry stage, it is extremely time consuming on querying and is prone to errors at this stage. Method a is fast and accurate on querying but is tedious and prone to errors on annotation. A compromise is required. The solution adopted in RAIDER is a content based annotation method which allows speed at both the annotation and retrieval stages.

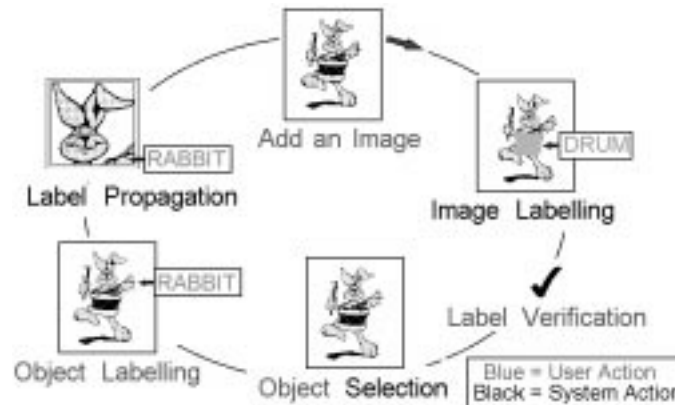


Figure 2: Illustration of Interactive Content Based Annotation

Figure 2 illustrates the process undertaken in RAIDER when an image is added to the database. The system first attempts to label all objects in the scene via the use of colour, shape, texture and their combinations. The labels are then verified by the user and deselected as necessary. The user selects and labels remaining objects via an interactive object selection process (detailed in Section 6). The label is propagated through the image (intra-frame propagation) and the rest of the database (inter-frame propagation) if required via the use of colour and texture classification and segmentation techniques. As time progresses RAIDER's knowledge increases, therefore the user's workload decreases. In more specialised databases (e.g. when all possible objects are known in advance) annotation can become a fully automatic process.

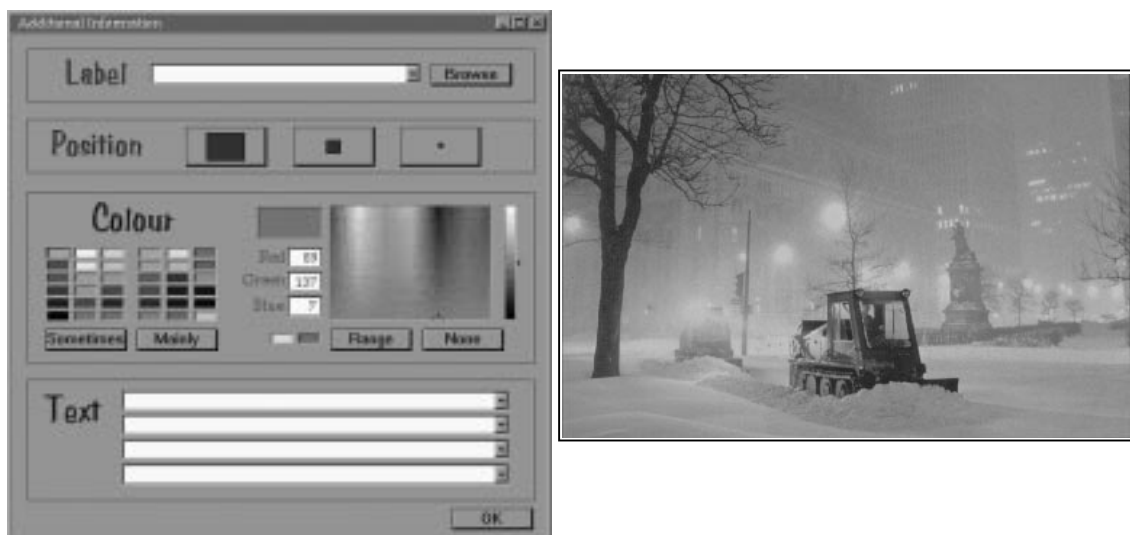


Figure 3: Additional Information Window and an Example of its use for Colour Specification

Figure 3 shows the additional information window of RAIDER. It is from here that objects are labelled during the object selection stage of the annotation process. Information which the system cannot derive from image data, for example names and dates, can be provided at this time. Optional information to assist RAIDER at the retrieval stage can also be given. Specifying the position of an object (close or distant) prevents confusion in texture analysis as an object's appearance changes with distance. The colour selector can be used to specify general object colours. For example if the snow in

Figure 3 is labelled at the annotation stage, RAIDER would believe that snow is pink. This is obviously incorrect and will effect the success of queries which rely on colour analysis (including similar image queries). The colour selector in the additional information window can be used to specify that snow is usually white thus increasing the overall retrieval accuracy of the system.

Image annotation is traditionally inadequate and tedious. RAIDER's annotation method combats both of these problems and the process is quick, easy and effective. The following section explains how the inadequacies of traditional annotation are addressed in RAIDER.

2.3 Introducing Flexibility in to Text Based Queries

In traditional systems an image can only be retrieved if it possesses an identical keyword to the query label (even if the object is present in the image). This problem can be overcome with an object hierarchy.

If an object is not known to the system but can be located in the hierarchy, a path through the graph nodes can lead to alternative search objects. For example, in Figure 4, the query search string is 'Petals' with which the system is unfamiliar. Traversing the hierarchy leads to 'Flowers' then 'Plants', a label known to the system, all images containing plants are returned.

This is a potentially explosive situation and a research topic in its own right. An implementation for specialised databases is however feasible and prevents exhaustive image labelling thus significantly speeding up the annotation stage.

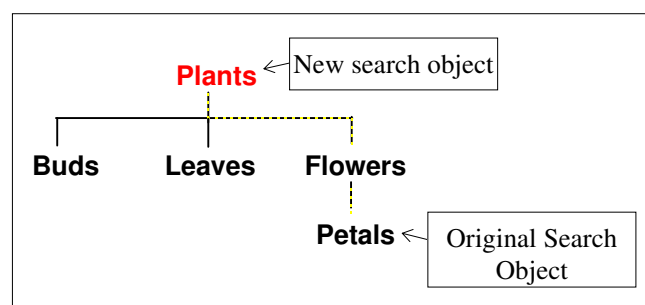


Figure 4: An Annotation Hierarchy Example

If a query object is unknown to the system and not present in the hierarchy the user can provide a sample image or define the object in terms of texture, colour and shape. RAIDER can analyse this information and perform a content based search on the database. The combination of a label hierarchy and content based searches ensure that the user is not restricted to object labels contained in the database. Flexibility has been introduced in to the query process.

2.4 RAIDER at a Lower Level

Two tables of information are maintained in RAIDER. The first contains information on individual objects and is referenced at the retrieval stage. The object id, parent image id, object feature vector and user added additional information are stored.

The second table contains a mean feature vector for each object type. This information is used when a new image is added to the database (the system automatically labels all known objects in the scene). It can also be used to search for objects in unlabelled images at the retrieval stage. The table is updated when a new object is located, for example, at the label propagation and user object labelling stages of the annotation process.

The success of a content based query depends on the quality of the feature vector fields. Texture analysis methods have been developed for use in RAIDER in an attempt to obtain appropriate features for inclusion in these fields. The following sections give an overview of the texture analysis techniques developed to date for use in RAIDER.

3 Rotation Invariant Texture Analysis

Texture analysis has been a major research area for decades. Many established methods exist for the classification of textured images. Unfortunately most techniques assume that the textures are uniformly presented and captured from the same viewpoint. This is an unrealistic assumption in the real world [6]. For applications such as content based image retrieval, texture analysis often need to be invariant to viewpoints. Genuine viewpoint invariance is extremely difficult to obtain. Rotation invariance (an important aspect of the general viewpoint invariance problem) is a practical starting position and forms the main goal of the studies in this paper. In this section two novel algorithms are described for extracting rotation invariant texture features for use in RAIDER.

3.1 The Multichannel Filtering Method

A multichannel filtering technique based on Gabor filters in the frequency domain is used to acquire rotation invariant texture features. The definition of a Gabor filter is given in Equation 1.

$$h(x, y) = g(x, y)e^{-2\pi j(u_o x + v_o y)} \quad (1)$$

where $g(x, y)$ is a Gaussian (assumed to be isotropic) of the form:

$$g(x, y) = e^{-\frac{1}{2} \left[\frac{x^2 + y^2}{\sigma^2} \right]} \quad (2)$$

This function can be split into two parts, the even and odd filters $h_e(x, y)$ and $h_o(x, y)$ which are also known as the symmetric and antisymmetric filters respectively. These filter pairs are given in Equation 3 and are used in the multichannel method of rotation invariant texture analysis.

$$h_e(x, y) = g(x, y) \cos(2\pi f(x \cos \theta + y \sin \theta))$$

$$h_o(x, y) = g(x, y) \sin(2\pi f(x \cos \theta + y \sin \theta)) \quad (3)$$

where: $f = \sqrt{u_o^2 + v_o^2}$ and $\theta = \arctan(v_o/u_o)$

The Fourier transform of the filters is taken and the output images obtained via FFT. For example:

$$q_e(x, y) = \text{FFT}^{-1}[P(u, v)H_e(u, v)] \quad (4)$$

where $P(u, v)$ is the Fourier transform of the input image $p(u, v)$ and $H_e(u, v)$ that of the filter $h_e(u, v)$. The outputs of the two filters are combined using the following equation to obtain a single value at each pixel (see [13-14] for a justification of this combination):

$$q(x, y) = \sqrt{q_e^2(x, y) + q_o^2(x, y)} \quad (5)$$

The two main filter parameters, which define the filter's location in the frequency domain, are the radial frequency (f) and the orientation (θ). For each radial frequency, filters are positioned at, and sampled around, a circle of radius f . $180/\Delta\theta$ filters are thus required per frequency as conjugate symmetry is exploited, where $\Delta\theta$ is the sampling interval. For a given frequency f the energy values of the filtered images form a periodic function of θ with period π . A rotation of the input image corresponds to a translation of this function. n rotation invariant features are obtained from the first n magnitudes of the periodic function's Fourier expansion.

The process is repeated for each of x frequencies resulting in an xn -dimensional feature vector which can be used during classification. Further details of the method may be found in [8]. Similar rotation invariant features are proposed in [4] and [9]. A sampling interval of 10° was used and 3 features were retained per radial frequency ($f=2, 4, 8, 16, 32, 64$) [7].

3.2 The Edge Attribute Method

In the second method a Sobel edge operator is used to generate gradient direction and magnitude images of the input texture. The gradient directions (α) at all pixels are then histogrammed and weighted by the corresponding gradient magnitudes. The resulting histograms are spiky; spurious spikes are removed by smoothing. Normalisation is required to remove the undesirable effects of different illuminations. The following equation defines the normalisation technique used:

$$B(\alpha) = \frac{b(\alpha)h}{m} \quad (6)$$

where h is the desired height of the histograms, m is the largest histogram value and $B(\alpha)$ and $b(\alpha)$ are the normalised and original values at a histogram bin α respectively.

The cyclic direction histogram formed can be regarded as a periodic function of α with period 2π where a rotation of the image results in a translation of this function. The Fourier transform of the periodic function is taken, the magnitudes of the function's Fourier coefficients are invariant to rotations; the first n magnitudes can be represented in an n -dimensional feature vector for use in classification.

3.3 Test Database

A test database was created for texture classification and image retrieval experiments. The database consists of 44 Brodatz [5] texture classes shown in Figure 5. Each texture was randomly rotated and cropped to 128*128 pixels. The resulting images were subjected to histogram equalisation to prevent bias towards images with similar grey levels. A total of 1320 images (30 from each texture class) were obtained.

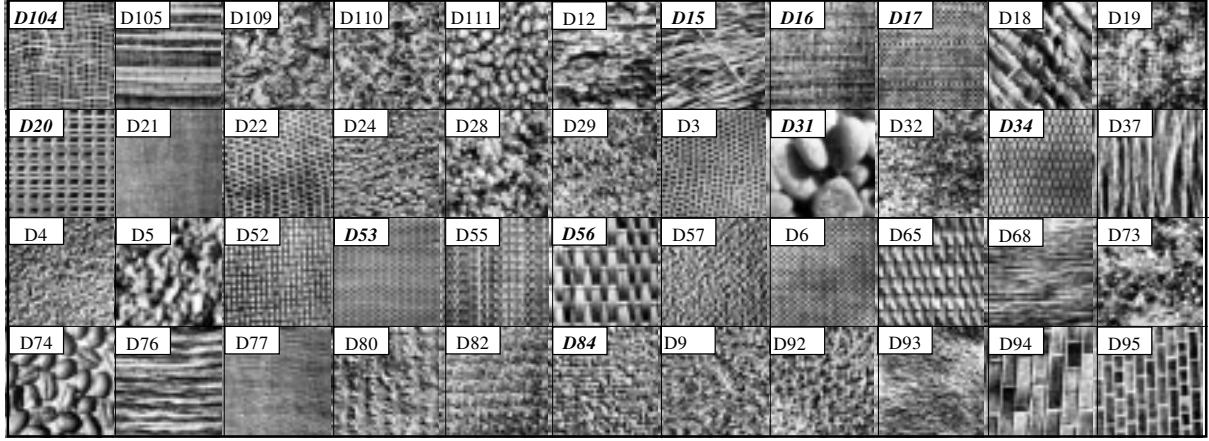


Figure 5: The 44 Brodatz Texture Classes Contained in The Test Database

3.4 Classification Experiments

Texture classification is required during both the annotation and retrieval sections of the RAIDER system. During the experiments half the database was used as training data, exemplar feature vectors were obtained by averaging the feature vectors for each texture class. The second half of the database was used during classification where the Euclidean distance classifier was used.

The multichannel filtering method achieved a 94% overall correct recognition rate and the edge attribute method a rate of 53%. A breakdown of this result into individual texture classes is given in Table 1.

Texture	Multi-Channel	Edge Attribute	Texture	Multi-Channel	Edge Attribute	Texture	Multi-Channel	Edge Attribute
D104	100	20	D28	100	0	D55	100	76.7
D105	73.3	70	D29	86.7	33.3	D56	96.7	83.3
D109	90	26.7	D3	100	33.3	D57	90	53.3
D110	90	60	D31	100	26.7	D74	96.7	50
D111	100	73.3	D32	100	40	D76	76.7	23.3
D12	73.3	50	D34	100	76.7	D77	100	23.3
D15	96.7	70	D37	73.3	66.7	D80	96.7	33.3
D16	100	33.3	D4	90	46.7	D82	100	13.3
D17	100	53.3	D5	100	60	D84	93.3	6.7
D18	96.7	60	D52	96.7	96.7	D9	83.3	33.3
D19	100	46.7	D53	100	80	D92	90	30
D20	93.3	83.3	D6	100	100	D93	96.7	33.3
D21	100	93.3	D65	96.7	83.3	D94	100	70
D22	100	86.7	D68	100	93.3	D95	93.3	80
D24	80	50	D73	96.7	20			

Table 1: Classification Rates for The Two Methods

3.5 Method Comparison

The edge attribute method was found to be less accurate than the multichannel filtering method. Its main attraction is simplicity and efficiency; its execution time being a fraction of the multichannel filtering methods. The edge attribute method is an automatic process requiring no input or tuning parameters. It is these issues which render it suitable for image database applications. In contrast the multichannel filtering method is highly accurate. 99% of all textures are correctly identified within three guesses compared to 81% for the edge attribute operator. Four input parameters are required in the multichannel method; σ , θ , the radial frequencies and a value for the number of features to be kept per frequency. Experimental evidence suggests that optimal values for these parameters can be established and used. The methods execution time can be decreased by applying the filters to the image in parallel.

4 Content Based Image Retrieval in RAIDER

Both methods were included in RAIDER and image retrieval experiments performed. The experiments are based on similar image properties, i.e. a query image is presented to RAIDER which returns the n most similar images from the database according to a criterion c . Textures from the database described in Section 3 were used as query images and were presented to RAIDER in turn. The closest five images from the data-base were returned per search. Euclidean distance was used as the similarity measure (i.e. the criterion c).

Using the multichannel method an average of 98% of the images returned by RAIDER are of the same Brodatz texture class as the query image. This compares to 63% for the edge attribute method. The averages are decomposed into individual texture classes in Table 2.

17 texture classes gave perfect retrieval results for the multichannel method, compared to 1 for the edge attribute method. D6 (a highly regular texture) proved to be the most successful class obtaining a combined 100% retrieval rate. Textures D21 and D68 were also very successful in both methods. The results for D105 are the lowest using the multichannel method but are surprisingly high using the edge attribute method.

Similar experiments were performed on a variety of human subjects of different backgrounds and ages. The images were randomly presented to prevent bias in the results due to the learning element incurred in such experiments. An average of 85% of all images returned belonged to the same texture class as the query image. It is realised from this comparison that the multichannel method of texture analysis is extremely accurate.

Texture	Multi-Channel	Edge Attribute	Texture	Multi-Channel	Edge Attribute	Texture	Multi-Channel	Edge Attribute
D104	94.8	51.6	D28	98	33.3	D65	98	84
D105	91	79.3	D29	96	51.3	D68	100	90
D109	96	36.7	D3	100	39.3	D73	100	36
D110	96	62	D31	100	50.7	D74	98	46
D111	100	62	D32	96.7	33.3	D76	94.7	68.7
D12	96.7	68.7	D34	100	81.3	D77	100	40.7
D15	99.3	80.7	D37	97.3	54.7	D80	100	46.7
D16	97.3	51.3	D4	96	91.3	D 82	98.7	30.7
D17	100	63.3	D5	100	60.7	D84	92	35.3
D18	99.3	46.7	D52	100	98	D9	96	40.7
D19	100	48	D53	100	80	D92	100	51.3
D20	95.3	82	D55	99.3	90.7	D93	98.7	63.3
D21	100	94.7	D56	96.7	86.7	D94	99.3	60
D22	100	80.7	D57	94	86	D95	96	75.3
D24	94	56	D6	100	100			

Table 2: The Mean Retrieval Results for each Texture Class

5 Resistance to Noise

Another practically important but often overlooked issue in image databases and texture analysis is the noise robustness of texture features. In this section we outline our studies on the noise robustness of the rotation

invariant features in the context of image retrieval. For this purpose various levels of Gaussian noise ($\sigma=0-90$) were added to each image in the database.

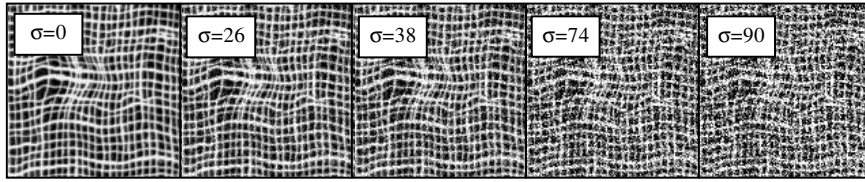


Figure 6: The Addition of Gaussian Noise to Texture D104

The resultant 6600 noisy images were then used to query the database. A total of five images were returned per search as before. Figure 6 shows texture D104 with the addition of various levels of noise.

Figure 7 shows the probability that an image returned from a search is of the same texture class as the query image. It can be seen that for the multichannel method, noise with a σ of 38 (shown in Figure 6) can be added to the query images before the retrieval rate drops equal to the edge attribute method using clean images. It is also at this noise level when the edge attribute curve begins to level off as retrieval becomes random. The shallower gradient of the multichannel filtering curve suggests that the method is more resistant to noise than the edge attribute method.

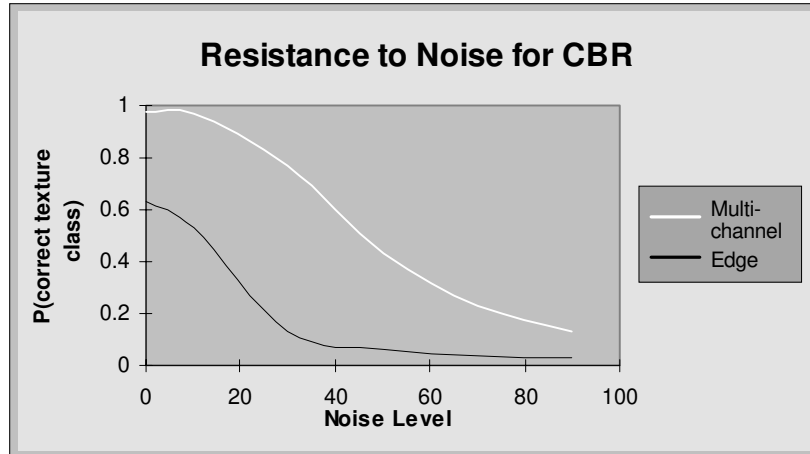


Figure 7: Results from The Multichannel and Edge Attribute Methods of CBR

It is interesting to note that for an average of one correct texture class returned per search the edge attribute and multichannel methods succeed to noise levels of 26 and 74 respectively. Example images containing such levels of noise are presented in Figure 6.

6 Object Selection

In the previous sections we have discussed rotation invariant texture features and their use in RAIDER for rotation invariant image retrieval. In this section we describe our initial work on content based image annotation in RAIDER. We focus on object selection as it is essential during annotation (intra- and inter-frame propagation). Manual object selection is a painstaking experience (especially for complicated objects such as trees which could take hours to outline with a mouse) and fully automatic object selection (image segmentation) is generally beyond the state-of-the-art in image processing and computer vision. Therefore a semi-automatic method has been developed. At present object selection is based either on colour or texture. In both cases the user draws a dragbox over the object to be labelled (see Figure 6) which is later extended automatically.

6.1 Multi-Channel Colour Segmentation

The HSV colour space is used in order to minimise errors due to brightness differences. A colour histogram (hue-saturation-number of pixels) of the marked area is compiled. A region growing technique is employed which takes the central pixel of the area as a seed. On region growing the smoothed hue and saturation values of the test pixel

are used to index the colour histogram. If the relevant bin contains entries the pixel is part of the region. Figure 8 shows object selection examples on natural scenes.

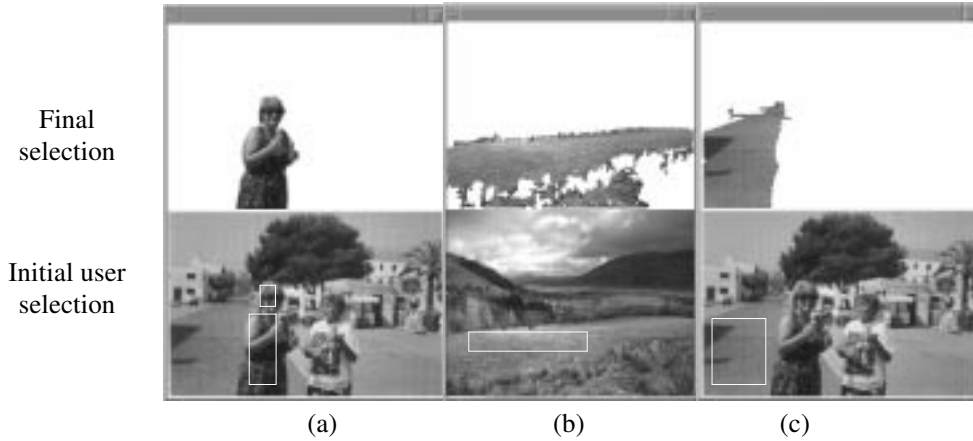


Figure 8: Object Selection using the Multi-Channel Colour Method

6.2 Rotation Invariant Texture Segmentation

Texture is often a more appropriate segmentation feature than colour. Texture analysis is required when colour segmentation fails. The method developed is based on multichannel Gabor filtering, as explained in Section 3.1, with the addition of a dynamic frequency detection stage. Filters must be positioned at areas of high activity, therefore, peaks in the image's power spectra are located and these x frequencies are selected along with a sampling angle of 10° for filter placement.

The filtered images are analysed and rotation invariant features extracted at each pixel. Object selection then continues as in Section 6.1 using the rotation invariant features. Examples are given in Figure 9 (one is a synthetic image and the other is a natural image showing a shirt flanked by a jacket). Images in Figure 9(a) are the original with user selected regions; those in (b) the final selection based on colour; and those in (c) the final selection based on texture. The results show that the use of colour fails to locate the desired object and that the texture based method is invariant to image rotation.

Once the entire image has been segmented and the objects located, attributes such as rotation invariant texture features discussed in Section 3 can then be computed for each segmented/located region/object. Such attributes will subsequently be used in label propagation.

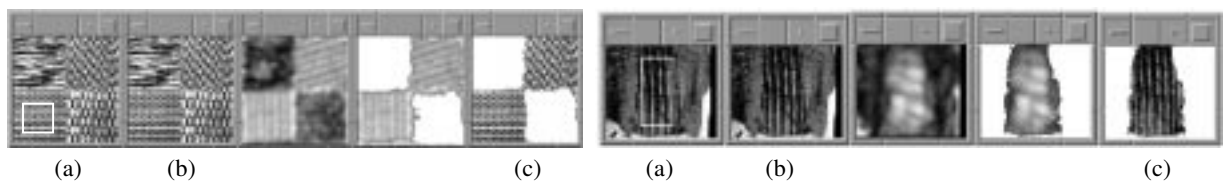


Figure 9: Object Selection Based on Texture Analysis

7 Conclusions

RAIDER has been introduced as an image database management system which exploits content based annotation and retrieval for increased flexibility and efficiency of database searches. An overview of the two main components (annotation and retrieval) of RAIDER was given. Two rotation invariant texture analysis techniques have been explained and classification results on a database of over 1300 images presented. Correct classification rates of 94% and 55% (99% and 83% using three returns) were obtained for the multichannel Gabor method and edge attribute methods respectively.

The techniques were incorporated in to RAIDER and retrieval experiments conducted. 98% and 63% of all textures returned from a search were of the correct texture class for the multichannel and edge attribute methods respectively. 6600 query images containing various levels of Gaussian noise were used to test the methods' resistance to noise. The noise level rose to a standard deviation of 26 and 74 for the edge attribute and

multichannel methods respectively before an average of only 1 correctly retrieved image out of 5 was reached (1 per search).

The multichannel method was also applied to object selection for use in the annotation process. Rotation invariant texture segmentation of both synthetic and natural images was obtained. A multi-channel colour histogramming method of segmentation was also successfully applied to object selection.

8 Future Work

The implementation of the RAIDER system is in its infancy and a great deal of work remains to be done. Future work includes geometric invariant texture analysis and the combination of texture and colour for image retrieval and label propagation.

9 References

1. M. Flickner et al. Query By Image and Video Content: The QBIC system. IEEE Computer 1995; 28:9
2. A. Pentland et al. Photobook: Content-Based Manipulation of Image Databases. M.I.T. Media Laboratory, Technical Report 1994; 255
3. R. Picard and T. Minka. Vision Texture for Annotation. Multimedia Systems 1995; 3:3-14
4. B. S. Manjunath et al. Rotation Invariant Texture Classification using Modified Gabor Filters. Proceedings of IEEE ICIP95 1995. 262-265
5. Brodatz. Textures: A Photographic Album for Artists and Designer. Dover, NY 1966
6. T. N. Tan. Geometric Invariant Texture Analysis. Procs SPIE 1995; 2488;475- 485
7. S. R. Fountain and T. N. Tan. Extraction of Noise Robust Rotation Invariant Texture Features Via Multichannel Filtering. Proceedings of IEEE ICIP97 1997; 3:197-200
8. T. N. Tan. Noise Robust and Rotation Invariant Texture Classification. Proceedings of EUSIPCO-94 1994; 377-1380
9. Greenspan et. al.. Rotation Invariant Texture Recognition using a Steerable Pyramid. Proceedings of ICPR94 1994; 162-167
10. S. C. Orphanoudakis et al. I²C: A system for The Indexing, Storage, and Retrieval of Medical Images by Content. Medical Information 1994; 19:2:102-122
11. H. Samet. MARCO: MAp Retrieval by Content. IEEE Transactions on Pattern Analysis and Machine Intelligence 1996; 18:8
12. M. Cascia and E. Ardizzone. JACOB: Just Another Content-Based Query System for Video Databases. Proceedings of ICASSP-96 1996; 7-10
13. D. A. Pollen and S. F. Ronner. Visual Cortical Neurons as Localised Spatial Frequency Filters. IEEE Transactions SMC 1983, 13:907-916
14. T. N. Tan. Texture Feature Extraction via Visual Cortical Channel Modelling. Proceedings of. ICPR 1992; C607-C610