

# Content Based Image Retrieval through Object Extraction and Querying

A.H.Kam, T.T.Ng, N.G.Kingsbury and W.J.Fitzgerald  
Signal Processing Laboratory  
University of Cambridge Engineering Department  
ahswk2, ttn21, ngk, wjf@eng.cam.ac.uk

## Abstract

We propose a content based image retrieval system based on object extraction through image segmentation. A general and powerful multiscale segmentation algorithm automates the segmentation process, the output of which is assigned novel colour and texture descriptors which are both efficient and effective. Query strategies consisting of a semi-automated and a fully automated mode are developed which are shown to produce good results. We then show the superiority of our approach over the global histogram approach which proves that the ability to access images at the level of objects is essential for CBIR.

## 1. Introduction

Image retrieval has traditionally been based on manual caption insertion describing the scene which can then be searched using keywords. Caption insertion is a very subjective procedure and quickly becomes extremely tedious and time consuming, especially for large image databases which are becoming ever more common with the growing availability of digital cameras and scanners. There is thus an urgent need for effective content-based image retrieval (CBIR) systems.

We believe the key to effective CBIR performance lies in the ability to access the image at the level of *objects*. This is because users generally want to search for images containing particular object(s) of interest and thus the ability to represent, index and query images at the level of objects is critical [3].

In this paper, we present a framework for CBIR based on unsupervised segmentation of images into classes and querying using properties of these classes. As these segmented classes are homogeneous in some sense (in our case, colour and texture), they correlate well with the identity of objects. By decomposing images as combinations of objects in this manner, querying becomes more meaningful and intuitive than it is with global image properties. This is



Figure 1. Decomposing an image by segmentation into classes corresponding to 'objects'

obviously true for images with distinct foreground objects but the rationale also holds for 'background' images where no interesting foreground objects are present. Images belonging to the latter category can be thought of consisting of combinations of classes with homogeneous colour and texture (for example, images of the seaside generally consist of the beach and the sea, images of sunset scenes generally consist of the reddish sky and dark silhouettes and so forth) and querying is made more effective by being based on these class combinations which characterise the scene.

In our CBIR implementation, images are firstly segmented based on joint colour and textural features using our previously developed unsupervised multiscale segmentation algorithm [6], [7]. The segmentation process is completely unsupervised and performed off-line for each image. Following this, we represent each image using effective and compact colour and textural descriptors of its classes. We then structure the descriptor database following a relational model which allows its implementation on powerful relational database engines. Class attribute queries are processed using a parallel strategy which results in significant speed-up in the retrieval process if parallel processor machines are used.

In Section 2, we will briefly describe the segmentation algorithm employed. We will then discuss the descriptors assigned to each class in Section 3. In Section 4, we present our query strategy as well as preliminary results from queries on our image database testbed consisting of various natural images.

## 2. Unsupervised Segmentation

Our unsupervised segmentation algorithm involves the following steps:

1. Normalised colour and texture features (three for colour and two for texture) are mapped to a multidimensional feature space. Spatial information is incorporated into the process by including spatial features into the feature space. The colour space used is S-CIE  $L^*a^*b^*$ , the spatial extension of the perceptual uniform CIE  $L^*a^*b^*$ , originally developed by Zhang and Wandell [10]. This colour space takes into account the appearance of fine-patterned colours on the human visual system. Textural features meanwhile are generated using the logarithm of the energies of the 2-D complex wavelet coefficients [8]<sup>1</sup> and taking the top two *principal components*.
2. Significant features which correspond to clusters in the feature space are assumed to be representations of underlying classes, the recovery of which is achieved using the *mean shift* procedure [4], a robust kernel based decomposition method. The kernel size used was fixed for all images, resulting in a decomposition into an appropriate number of classes for each image.
3. By determining the number of classes and the properties of each class via step 2, a Bayesian multiscale processing approach, which models the inherent uncertainty in the joint specification of class and position spaces using the Multiscale Random Field model [1], is used for the subsequent classification process.

Typical segmentation maps of images in our database are shown in Figure 2.

## 3. Describing the Classes

Once an image has been segmented, we proceed to extract a description of each class with the total description of classes constituting a description of the image. A class descriptor has to embody the class characteristics (which typically translates to representing a particular object) in an effective fashion to facilitate efficient indexing and accurate retrieval. Thus, designing an effective class descriptor is more difficult than designing feature extractors for segmentation and thus they should be seen as separate processes.

<sup>1</sup>For more information about the 2-D complex wavelet transform, please visit: <http://www-sigproc.eng.cam.ac.uk/ngk>

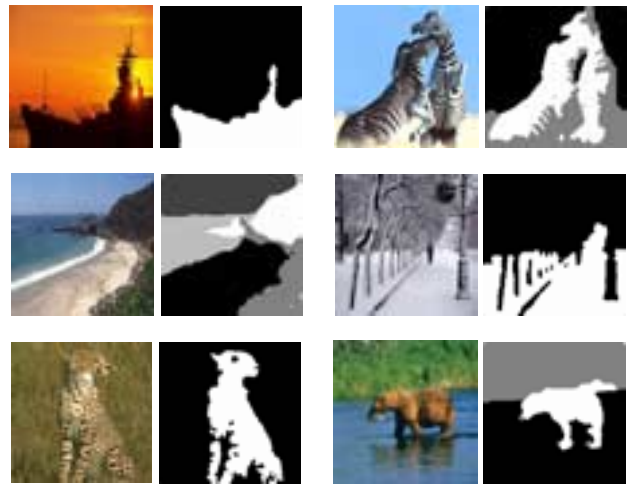


Figure 2. Typical segmentation maps of images

### 3.1. Colour Descriptors

In order to represent the colour distribution of each class, we store the colour histograms of the pixels of the class. This histogram is based on bins with width 10 in each dimension of the S-CIE  $L^*a^*b^*$  colour space. This spacing yields 10 bins in the  $L^*$  dimension and 40 bins in each of the  $a^*$  and  $b^*$  dimensions, for a total of 90 numbers as colour descriptors.

To evaluate the dissimilarity between the colour histograms of two classes/objects, we apply the *Kolmogorov-Smirnov* (K-S) distance, as originally proposed in [5]. The K-S distance essentially measures the difference between two probability distribution function. If  $F_1(k)$  and  $F_2(k)$  are two independent sample distribution functions (i.e. histograms) defined such that:

$$F_t(k) = \frac{1}{n} \#(i : y_i^t \leq k) \quad (1)$$

where  $n$  is the number of data samples,  $y_i^t$  so that  $1 \leq i \leq n$ , then the K-S distance is the maximum difference between the distribution over all  $k$ :

$$\text{K-S}(y^1, y^2) = \max |F_1(k) - F_2(k)| \quad (2)$$

The overall colour dissimilarity measure between two classes with colour histograms  $\mathbf{x}^{COL}$  and  $\mathbf{y}^{COL}$  is taken to be the root mean square of the K-S distances of each of the  $L^*$ ,  $a^*$  and  $b^*$  histograms:

$$d_{COL}(\mathbf{x}^{COL}, \mathbf{y}^{COL}) = \frac{1}{3} \{ [\text{K-S}(\mathbf{x}_{L^*}^{COL}, \mathbf{y}_{L^*}^{COL})]^2 + [\text{K-S}(\mathbf{x}_{a^*}^{COL}, \mathbf{y}_{a^*}^{COL})]^2 + [\text{K-S}(\mathbf{x}_{b^*}^{COL}, \mathbf{y}_{b^*}^{COL})]^2 \}^{\frac{1}{2}} \quad (3)$$

As the range of K-S distances lie between 0 and 1, the colour similarity measure,  $s_{COL}(\mathbf{x}^{COL}, \mathbf{y}^{COL})$  is simply taken as:

$$s_{COL}(\mathbf{x}, \mathbf{y}) = 1 - d_{COL}(\mathbf{x}, \mathbf{y}) \quad (4)$$

### 3.2. Texture Descriptors

For each class, texture is described by the distribution of the magnitude of its complex wavelet coefficients,  $f(\mathbf{x}^{TEX})$ . Using four levels of the 2-D complex wavelet transform (which yields six complex subbands at every level) produces a total of 24 subbands, the magnitude of each is converted into a histogram and modelled as a generalised Rayleigh distribution (see figure 3):

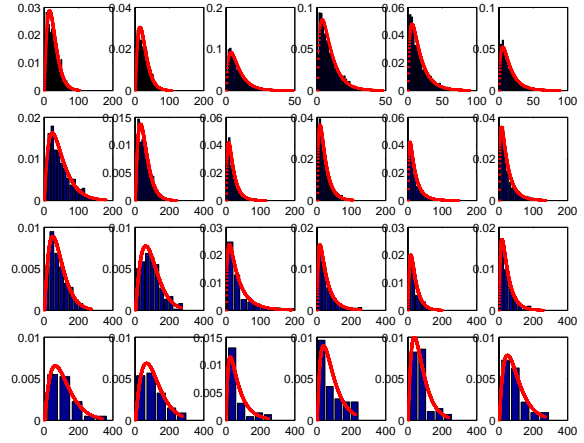
$$f(x_i^{TEX}) = k_i x_i^{TEX} \exp - \left( \frac{x_i^{TEX}}{\sigma_i} \right)^{\beta_i} ; i = 1, 2, \dots, 24 \quad (5)$$

where, to achieve the same mean and variance as the input sample distribution:

- $k_i = \frac{\beta_i}{\sigma_i^2 \Gamma(\frac{2}{\beta_i})}$  where  $\Gamma(\cdot)$  denotes the gamma function
- $\sigma_i = \frac{m_1 \Gamma(\frac{2}{\beta_i})}{\Gamma(\frac{3}{\beta_i})}$  where  $m_1 = E(x_i^{TEX})$
- $\beta_i = F^{-1} \left( \frac{m_1^2}{m_2} \right)$  where  $m_2 = E[(x_i^{TEX})^2]$  and  $F(x) = \frac{[\Gamma(\frac{3}{x})]^2}{\Gamma(\frac{2}{x})\Gamma(\frac{4}{x})}$

Thus, for each class, the generalised Rayleigh model parameters,  $\sigma_i$  and  $\beta_i$  is calculated for each of the 24 histograms, for a total of 48 numbers as texture descriptors. To compute the texture dissimilarity between two classes, we:

1. Generate probability distribution functions for each of the 24 subbands of each class using the stored values of  $\sigma_i$  and  $\beta_i$ .
2. Apply the K-S distance between histograms corresponding to the same subband.
3. The overall texture dissimilarity measure is calculated as the root mean square of all the K-S distances. The final similarity measure is given by the subtraction of the dissimilarity measure from unity.



**Figure 3. Top row: Image of a typical tiger and extraction of the class corresponding to the creature. Bottom row: Histograms of the magnitude of the tiger’s complex wavelet coefficients (every row depicts a decomposition level with level 1 at the top) and the modelling performance of the generalised Rayleigh distribution (plotted in red)**

## 4. Image Retrieval

The class descriptor database is structured using a relational model. This allows its implementation on powerful commercial relational database engines and for queries and retrieval to be described using SQL’s *Select* and *Join* operations [9]. For example, as the first step, descriptors of particular classes of an image can be extracted from the database using a simple *Select* operation.

### 4.1. Querying Strategy

There are two modes of operations for our image retrieval system: a semi-automated mode and a fully automated mode. In the semi-automated mode, the user composes a query by submitting an image and by seeing the segmentation map, selects the class or classes to match. There is also an option of selecting the relative importance of the classes (should there be more than one in the query composition); by default, all classes in a query are considered

equally important.

All ‘compound’ queries, i.e. queries being based on more than one class, are firstly decomposed into ‘simple’ queries, i.e. queries based on a single class. The similarity match for each simple query is calculated as follows:

1. Colour and texture descriptors for the queried class are retrieved from the descriptor database
2. The similarity measures for colour and texture are computed for classes in the database whose sizes (specified as a fraction of the image) are at least 25% of the queried class
3. The overall similarity measure is taken to be the weighted combination of the colour and texture similarity measures, with the weights set by the user. By default, colour and texture similarities are weighted equally

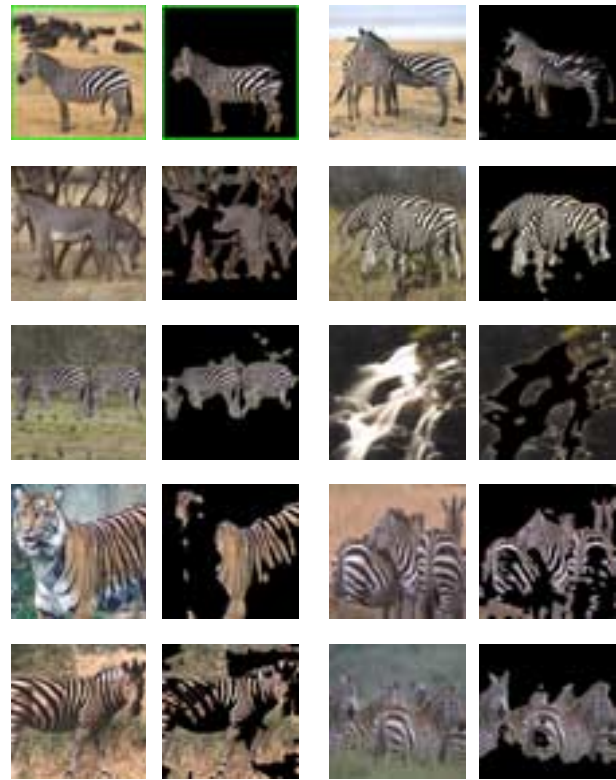
The SQL *Join* operation on the simple queries’ match lists will obtain the set of common images, with the best match maximising the similarity measures, weighted according to their relative importance. As simple queries can be processed in parallel, significant speed-up in the retrieval process is possible with parallel processor machines.

In the fully automated mode, the user has only to submit a query image and the algorithm is designed to handle the rest. In this case, we first perform simple queries on classes of the image which constitute at least 10% of the image. In the absence of a theoretical foundation to determine the relative importance of the classes, we simply sum up the top 10 similarity measures of the match lists of each of these classes. This step will provide us information as to which classes have relatively high matching scores and thus possess a higher probability of being an ‘object of interest’. Finally, a compound query is performed on the top two classes of the image with the highest matching scores.

## 4.2. Results

We have performed a variety of queries on our small image database testbed for both the semi-automated and the fully automated mode. Preliminary results are encouraging as shown in Figures 4 and 5. We are currently in the process of expanding our image database to include as many varied natural images as possible.

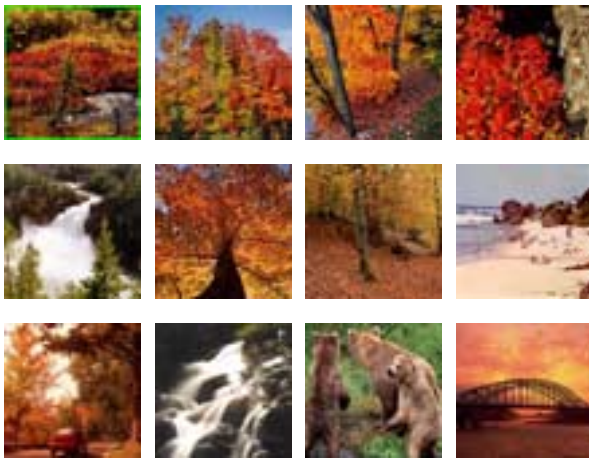
Figure 6 depicts retrieval precision and recall performance of several image categories of our system. Results were generated using a one-class default query for zebra and tiger images while automated retrieval was utilised for sunset and autumn scene images where no interesting foreground objects are present. Retrieval performance is particularly good for sunset and zebra images while results for tiger and autumn scene images aren’t too bad either.



**Figure 4. Example of a one-class default query: Query image and the selected class (with green borders, top left), with the retrieved images, depicted with the matching class, arranged from highest similarity, from left to right, top to bottom**

We also compared the performance of our method with and without the pre-segmentation stage (i.e. in the latter case, querying based on global histograms). For fair comparison, the fully automated mode is used for the approach with the pre-segmentation stage.

The table below compares the image retrieval rates between the two approaches for leopard, bear, sunset and winter scene images. The approach with the pre-segmentation stage performs better for all image categories tested although the global histogram approach produces reasonable results especially for sunset images. These results are consistent with our belief that the key to effective CBIR performance lies in the ability to access images at the level of objects.



**Figure 5. Example of an automated retrieval: Query image (with green borders, top left) and the retrieved images, arranged from highest similarity, from left to right, top to bottom**

Image Categories	Precision values based on the top 15 images returned	
	Pre-Segmented	Global Histogram
Leopard	40%	33%
Bear	27%	13%
Sunset	93%	87%
Winter	78%	67%

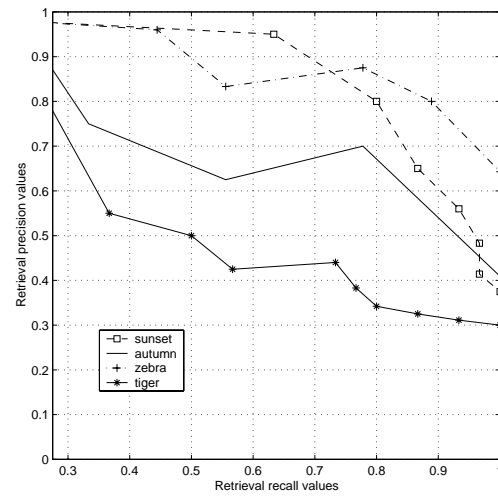
## 5. Discussion

We have proposed a content based image retrieval system based on classes of pre-segmented images. A general and powerful multiscale segmentation algorithm automates the segmentation process, the output of which is assigned novel colour and texture descriptors which are both efficient and effective. We then discussed our query strategy, for both the semi-automated and fully automated mode and demonstrated its encouraging results.

Nevertheless, low level features like colour and texture are generally insufficient for effective retrieval of unconstrained imagery. We are currently in the process of incorporating high-level descriptors into our method by training our system using *support vector machines* [2]. We believe this inclusion of semantics into the query and retrieval process will be the key to successful CBIR in the future.

## References

[1] C. Bouman and M. Shapiro. A Multiscale Random Field Model for Bayesian Image Segmentation. *IEEE Transactions on Image Processing*, 3(2):162–177, 1994.



**Figure 6. Precision vs. recall performance**

[2] C. Burges. A Tutorial on Support Vector Machines for Pattern Recognition. *Data Mining and Knowledge Discovery*, 2(2), 1998.

[3] C. Carson, M. Thomas, S. Belongie, J. Hellerstein, and J. Malik. Blobworld: A System for Region-based Image Indexing and Retrieval. In *Proceedings of the 3rd International Conference on Visual Information Systems*, 1999.

[4] K. Fukunaga and L. Hosteler. The Estimation of the Gradient of a Density Function, with Applications in Pattern Recognition. *IEEE Transactions on Information Theory*, 21:32–40, 1975.

[5] C. Graffigne, D. Geman, S. Geman, and P. Dong. Boundary Detection by Constrained Optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):609–628, 1990.

[6] A. Kam and W. Fitzgerald. Image Segmentation: An Unsupervised Multiscale Approach. In *Proceedings of the 3rd International Conference on Computer Vision, Pattern Recognition and Image Processing (CVPRIP 2000)*, volume 2, pages 54–57, Atlantic City, New Jersey, USA, February 27–March 3, 2000.

[7] A. Kam and W. Fitzgerald. General Unsupervised Multiscale Segmentation of Images. In *Proceedings of the 33rd Asilomar Conference on Signals, Computers and Systems*, Pacific Grove, California, USA, October 24–27, 1999.

[8] N. Kingsbury. Shift Invariant Properties of the Dual-Tree Complex Wavelet Transform. In *Proceedings of the IEEE Conference on Acoustics, Speech and Signal Processing (ICASSP'99)*, Phoenix, Arizona, USA, 1999. paper SPTM 3.6.

[9] J. Smith. *Integrated Spatial and Feature Image Systems: Retrieval, Compression and Analysis*. PhD thesis, Graduate School of Arts and Sciences, Columbia University, USA, February, 1997.

[10] X. Zhang and B. Wandell. Color Image Fidelity Metrics Evaluated Using Image Distortion Maps. *Signal Processing*, 70:201–214, 1998.