CBIR with Perceptual Region Features

Majid Mirmehdi and Radhakrishnan Periasamy
Department of Computer Science
University of Bristol
Bristol, BS8 1UB, UK
majid@cs.bris.ac.uk

Abstract

A perceptual approach to generating features for use in indexing and retrieving images is described. Salient regions that immediately attract the eye are colour (textured) regions that usually dominate an image. Features derived from these will allow search for images that are similar perceptually. We compute colour features and Gabor colour texture features on regions identified from a coarse representation of the image, generated by a multiband smoothing algorithm based on human psychophysical measurements of colour appearance. Images are retrieved, using a multi-feedback retrieval and ranking mechanism. We examine the performance of the features and the feedback mechanism.

1 Introduction

Textual annotation still remains a highly accurate and popular form of indexing for image database retrieval. However, it is a cumbersome and impractical task when a significant store of images must be produced, maintained, and updated. As an alternative approach, content based image retrieval (CBIR) systems aim to generate features automatically from images to provide a fast mechanism for indexing and retrieval. There is a large body of work in CBIR, including systems such as QBIC[1], VisualSEEk[2], and Photobook[3], reviewed in [4, 5], which has grown in recent years particularly encouraged by the fast global expansion of the internet and users' multimedia needs. Most CBIR systems use a variety of indexing features such as colour, texture, and shape. Other types of features sometimes used in CBIR are stated in [4] as wavelets or multiscale Gaussian derivatives.

In this work, we concentrate on the use of colour and colour texture features but based on perceptual appearance. The basic human perceptual categories of colour can be used as vital information to classify or segment colours [6]. While most CBIR methods using colour features rely on colour histogram indexing, the colour granularity provided by histograms is not always necessary especially when the final observer is a human. It is more natural to segment an image into regions of similar colour and use them to retrieve other images containing regions of similar colours. For example, Berlin and Kay [7] identified that humans segment colour into 11 basic categories, three achromatic (black, white, grey) and eight chromatic (red, green, yellow, blue, purple, orange, pink and brown). Such kinds of perceptual categories have been used in CBIR systems such as Perceptual Image Similarity And Retrieval Online (Pisaro) [8] which concluded that perceptual colour segmentation performs better than non perceptual-based techniques. De Bonet and Viola [9]

BMVC 2001 doi:10.5244/C.15.53

also approximated perceived visual similarity using a characteristic signature computed by extracting thousands of very specific local texture and global structure features.

The method proposed here is based on recent research on the psychophysics of the human visual system (HVS) [10]. Human colour perception, depends on the spatial frequency of the colour component. Thus, colours that appear in a multicolour pattern are perceived differently from colours that form uniform areas. For example, any coloured pattern with frequency higher than 8 cycles per 1° of visual angle is seen as black [10]. This concept was used in [11] as follows: they first generated a multiscale representation of a colour texture image using multiband Gaussian filters that emulated human colour perception and resulted in same size images representing what the human viewer would see at different distances from the scene. The coarsest image was then initially segmented as an embryonic, perceptual representation of the original image. Next, they developed a multilevel probabilistic relaxation algorithm to progressively improve and refine the initial segmentation while stepping through the multiscale images from coarse to fine.

In this work we are interested in, and use only partially, the perceptual smoothing stage of [11]. We assume that the regions in just one “distant” or coarse image adequately provide us with enough perceptual clues regarding the main colours and patterns of the image, and further refinement of an initial segmentation can be ignored. This is supported by the psychophysical evidence presented in [6]. We derive simple colour and texture features from each region found. The latter features are generated using the outputs of filters from a Gabor filter bank similar to [12, 13]. The use of Gabor filters is inspired by the multichannel filtering theory of visual information in the early stages of the HVS.

2 System Overview

In our CBIR system, we extract *perceptual* colour features and colour texture features to describe the characteristics of perceptually derived regions in the image. A feature vector is formed for each region. The distance measure between feature vectors is the Euclidean L_2 norm. We apply a recursive matching process which feeds back to the latest set of images found and selects a new subset to refine the rank of the final results without user intervention. Consider that we have a total of F images in the database. When processing a query, the query vector $\vec{q}$ is matched against all other feature vectors in the database and the G highest ranked images are selected, i.e. $G = \Psi(F; \vec{q})$ where Ψ is the query search operator. The feature vector $\vec{r}$ associated with a region spatially closest to the region from which the original query vector $\vec{q}$ came from is then selected and a new search is initiated. The new search however is limited to the set of G images whose feature vectors were matched from the initial search. The purpose is that only those subset H of G images are now ranked that contain regions with feature vectors that closely match both $\vec{q}$ and $\vec{r}$:

$$H = \Psi(G; \vec{r}) \quad \text{which is equivalent to} \quad H = \Psi(\Psi(F; \vec{q}); \vec{r}) \tag{1}$$

This process can be repeated again using the next spatially close region (with feature vector, say $\vec{s}$) and so on until there are no more regions in the query image. In our implementation, we perform this iteration three times by setting $G = 30$ and $H = 20$ until a final set of 10 images remain, however, the system could be easily adapted to perform more iterations. Three iterations is a good compromise between accuracy and speed. Two regions in an image are determined to be spatially close by comparing the proximity of their centres of gravity. Figure 1 illustrates a three step multi-feedback process. The multi-feedback subset retrieval and ranking process ensures that the final selection of images at the last step will be ranked to be as close as possible to the image the query vector came from, in terms of the number of common regions and overall similarity. It is automatic multi-region search and avoids user definitions and intervention.

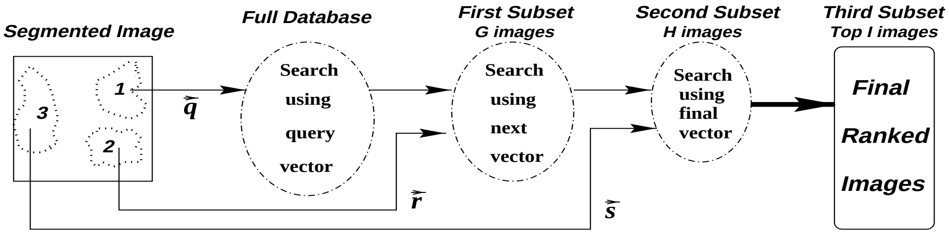


Figure 1: The multi-feedback subset retrieval and ranking process recursively ranks subsets of images from the full database. In this work: $G = 30$, $H = 20$, and $I = 10$.

3 Perceptual Smoothing and Feature Generation

The factors that influence the response characteristics of the HVS are the temporal and spatial variations of the stimuli as well as the spectral properties of the stimuli. When an observer deals with multi-coloured objects, their colour matching behaviour is affected by the spatial properties of the observed pattern [10]. Furthermore, the HVS will experience loss of detail at increasing distances away from the object. It perceives coloured textures at a large distance as areas of fairly uniform colour, whereas variations in luminance, e.g. at the borders between two textures, are still perceived. A multiscale smoothing algorithm was introduced in [11] that coarsens an image according to human perception depending on the distance it is being viewed from. The opponent colour space was used with each plane smoothed separately with different 2D spatial kernels [11], with the result that the luminance plane is blurred lightly, whereas the chromaticity planes are blurred more strongly. This spatial processing technique is pattern-colour separable. We implemented the 2D Gaussian kernels using 1D separable filters. Three typical filters used for the opponent colour bands are shown in Figure 2 to illustrate the varied amount of smoothing.

Once the kernels are generated and applied to the image in the opponent colour space, the image can be converted to CIE-Luv which is a perceptually uniform space and therefore more suitable for carrying out colour measurements and for generating representative features for our CBIR application. Figure 3 shows an image and its associated smoothed images at varying distances for both perceptual and typical single-kernel Gaussian smoothing. The Gaussian kernel used for comparison has the same size as the perceptual masks with σ being such that at the cutoff size its value is 1% of its central value. It is clear from Figure 3 that the perceptually smoothed images provide a more realistic representation and blurring of a scene viewed at varying distances than the Gaussian.

In [11] it was shown that a distance of 8 meters is a suitable choice for achieving good segmentation. In this work we apply the appropriate filters to obtain a smoothed image as if viewed at $8m$. This coarse image is then segmented using a customised K-means clustering method tuned to produce 3 to 5 clusters. The reason for this choice is that it

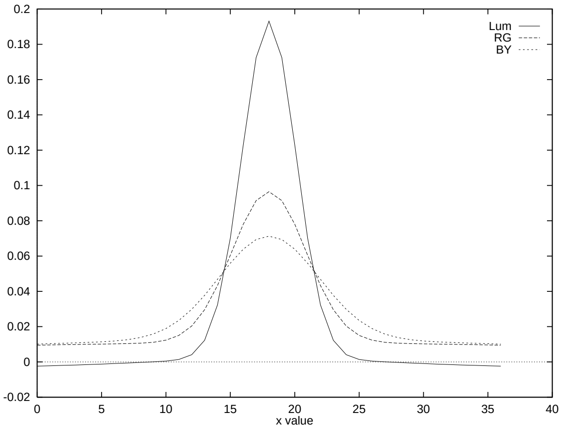


Figure 2: 1D view of combined Gaussian masks for the separate colour bands.

represents a reasonable number of main regions that a human observer may perceive when viewing an image. The segmented clusters are then restructured by rejecting some of their pixels in order to isolate *core clusters*, i.e. patches in which the pixels are definitely associated with the same region. The purpose of this is to remove noisy pixels from the region and reduce the risk of inaccurate features. To derive core clusters from the initial clusters, we need to fuzzify the classification result obtained after the K-means segmentation of our $8m$ blurred image[11]. We first compute the standard deviation σ_c of each cluster region $\Re_c, c = 1, .., C$ given C clusters. Then, we assign to each pixel i, a confidence, $\hat{p}_c^i$, with which it may be associated with each cluster:

$$\hat{p}_c^i \equiv \frac{\frac{\sigma_c^2}{d_c^{i^2}+\sigma_c^2}}{\sum_{k=1}^{C} \frac{\sigma_k^2}{d_k^{i^2}+\sigma_k^2}} \quad \forall i, \forall c \tag{2}$$

where $d_c^{i^2}$ is the squared distance of pixel i from the mean of cluster $\Re_c$ in Luv colour space. Each core cluster is formed from the pixels that can be associated with it with a confidence of at least 80%, i.e. $i \varepsilon \Re_c$ iff $\hat{p}_c^i \geq 0.8$. Figure 4 shows an original image, its $8m$ smoothed image and both the initial clusters and then the derived core clusters. The reduction of the original clusters into core clusters fulfils our goal of extracting a smaller representative group of pixel, coherent in both their colour and local texture properties.

3.1 Extracting Colour and Colour Texture Features

To compute cluster region statistics we use those pixel values in the original non-smoothed image whose addresses correspond to the core cluster locations in the $8m$ smoothed image. It is important not to use the pixels of the $8m$ smoothed image directly because the blurring of a coloured object in one image may not match the blurring of a similar colour object in another image due to differences in the colours in the neighbourhood of those objects in their respective images. We ignore the luminance band to remove the effects of variable illumination and use only the chromaticity channels u and v. The mean and variance of each region in each band is computed and stored in a 4-component colour

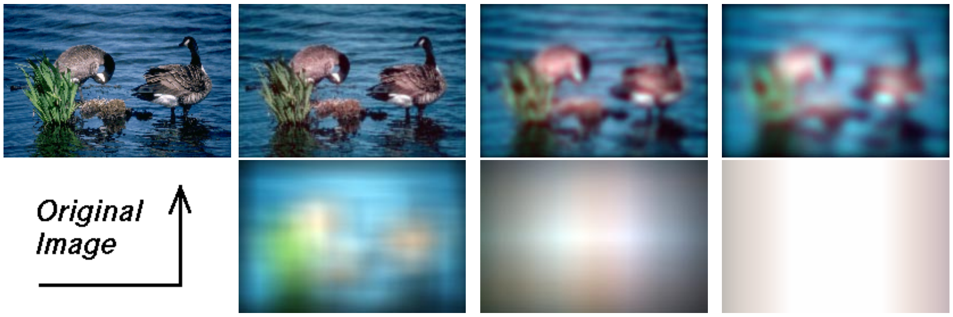


Figure 3: A real image and its perceptual (top) and Gaussian smoothing (bottom) corresponding to 1, 4, and 8 meters of viewing distance, from left to right.

feature vector. Hence, given $S(x, y)$ as the $8m$ smoothed image with C cluster regions, and $I(x, y)$ as the original image, then:

$$\mu_{uc} = \int\int^{\Re_c \varepsilon S(x,y)} I(x,y)dxdy \quad , \quad \sigma^2_{uc} = \int\int^{\Re_c \varepsilon S(x,y)} (I(x,y) - \mu_{uc})^2 dxdy \quad (3)$$

Similarly for the v chromaticity band, we obtain $[\mu_{vc}, \sigma^2_{vc}]$ resulting in the colour feature vector $\vec{f}_{uvc} = [\mu_{uc}, \sigma^2_{uc}, \mu_{vc}, \sigma^2_{vc}]$.

However, colour features alone are not enough to result in the desired retrievals. A query made using a region representing a green car should not return images of trees. We augment our colour features by the discriminatory power of texture. Gabor filters have been widely used as a model of texture for image interpretation [12]. They are designed to sample the entire frequency domain of an image by varying the bandwidth, shape, centre frequency and orientation parameters. In the spatial domain, a Gabor filter takes the form of a complex sinusoidal grating oriented in a particular direction and modulated within a Gaussian envelope. The convolution filter located at (x_0, y_0) with centre frequency, ω_0, orientation with respect to the x-axis, θ_0, and scales of the Gaussian's major and minor axes, α, β, is defined by [14]:

$$g_{(x,y)} = e^{-\pi\left[(x-x_0)^2/\alpha^2 + (y-y_0)^2/\beta^2\right]} \, e^{-2\pi i[u_0(x-x_0)+v_0(y-y_0)]} \quad (4)$$

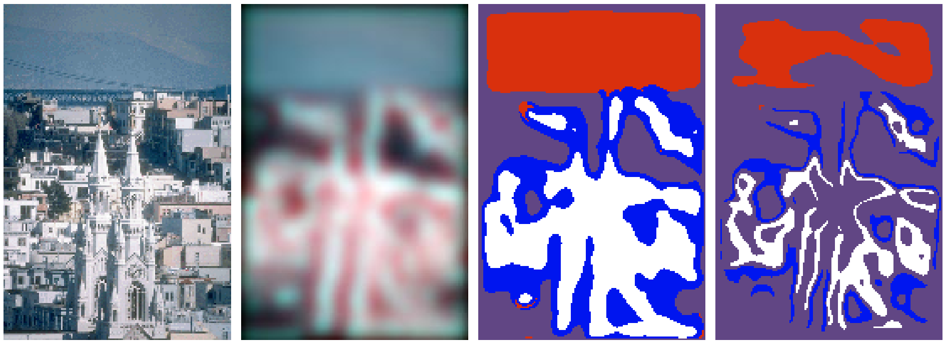


Figure 4: Left to right - original image, perceptually smoothed at $8m$, segmented into 4 clusters, and finally core clusters of the segmented image at 80% confidence.

This filter has a modulation of (u_0, v_0) such that $\omega_0 = \sqrt{u_0^2 + v_0^2}$ and the orientation of the filter is $\theta_0 = \tan^{-1}(v_0/u_0)$. Thus, each Gabor filter is tuned to detect only a specific local sinusoidal pattern of frequency ω_0, orientated at an angle of θ_0 in the image plane. The Gabor bank of filters applied in this study contain 16 filters: four different frequencies at 0.36, 0.18, 0.09, and 0.05 cycles/pixel, and four different directions at $0°$, $45°$, $90°$, and $135°$. The transform is performed as multiplication in the frequency domain. We take the mean (μ_{Guc}, μ_{Gvc}) and variance $(\sigma^2_{Guc}, \sigma^2_{Gvc})$ responses over each image region c in chromatic planes uv as our Gabor colour texture features in vector $\vec{f}_{Gc}$. Since there are 16 filters and two colour planes, we will have a 64 element feature vector per region:

$$\vec{f}_{Gc} = [\mu_{Guc_1}, \sigma^2_{Guc_1}, \mu_{Gvc_1}, \sigma^2_{Gvc_1}, ..., \mu_{Guc_{16}}, \sigma^2_{Guc_{16}}, \mu_{Gvc_{16}}, \sigma^2_{Gvc_{16}}] \tag{5}$$

To reduce the dimensionality of the feature vector we perform principal component analysis on the 64 Gabor features, similar to [13], and reduce them to 16 only (call them $\vec{f}_{gc}$) which represent 98% of the variance within the 64 dimensions. We then fuse the colour and colour texture feature vectors to form $\vec{f}_c$ as our final 20 component feature vector for the cluster region c, i.e. $\vec{f}_c = \vec{f}_{uvc} \cup \vec{f}_{gc}$. As the feature vector components come from different parameters, we also normalise the values to lie between 0 and 1.

4 Experiments and Results

Here we present the results of typical image queries using our fused feature vector $\vec{f}_c$. We also examine the performance of the perceptual features when used exclusively, i.e. $\vec{f}_{uvc}$ and $\vec{f}_{gc}$ acting on their own. Finally, we compare the correct number of retrieved images with and without using the multi-feedback retrieval and ranking process. A major hurdle in comparing one CBIR work against another is the difficulty in testing them on the same database. We report the accuracy of our results by considering the number of correctly retrieved images (i.e. they contain objects similar to the query judged subjectively) in the final set of 10 images. This basic measure is similar to that used in Pisaro [8].

Our database consists of 280 images of mainly outdoor scenes of buildings, people, animals, forests, and so on. These are segmented into 3 to 5 regions resulting in a total of 1061 regions. Features are extracted and stored in the database for all the regions. As new images are added, their feature vectors are simply appended to the existing feature set.

4.1 Retrieval Performance of Perceptual Features

A search consists of the comparison of the query vector against all other feature vectors in the database using the L_2 measure and then sorting them in order. Figure 5 shows the results of three queries and the best 10 ranked results, including the query image itself as the first image as expected. In each case we obtain 9 out of 10 images which are subjectively similar. In the “people” search we obtain an image of leaves in the 10th position which seems to be very similar in colouring to the other images. The same is true for the building image that appears in the “leaves” query. The most strange retrieval is the two elderly ladies in the “sky” query for which we have not yet found a clear explanation. The results overall show 90% accuracy in retrieving images that are of direct relation to the query. In all our experiments we had 7 to 9 correct images out of 10 giving an average hit rate of 80% which compares extremely well with other works[4, 5].

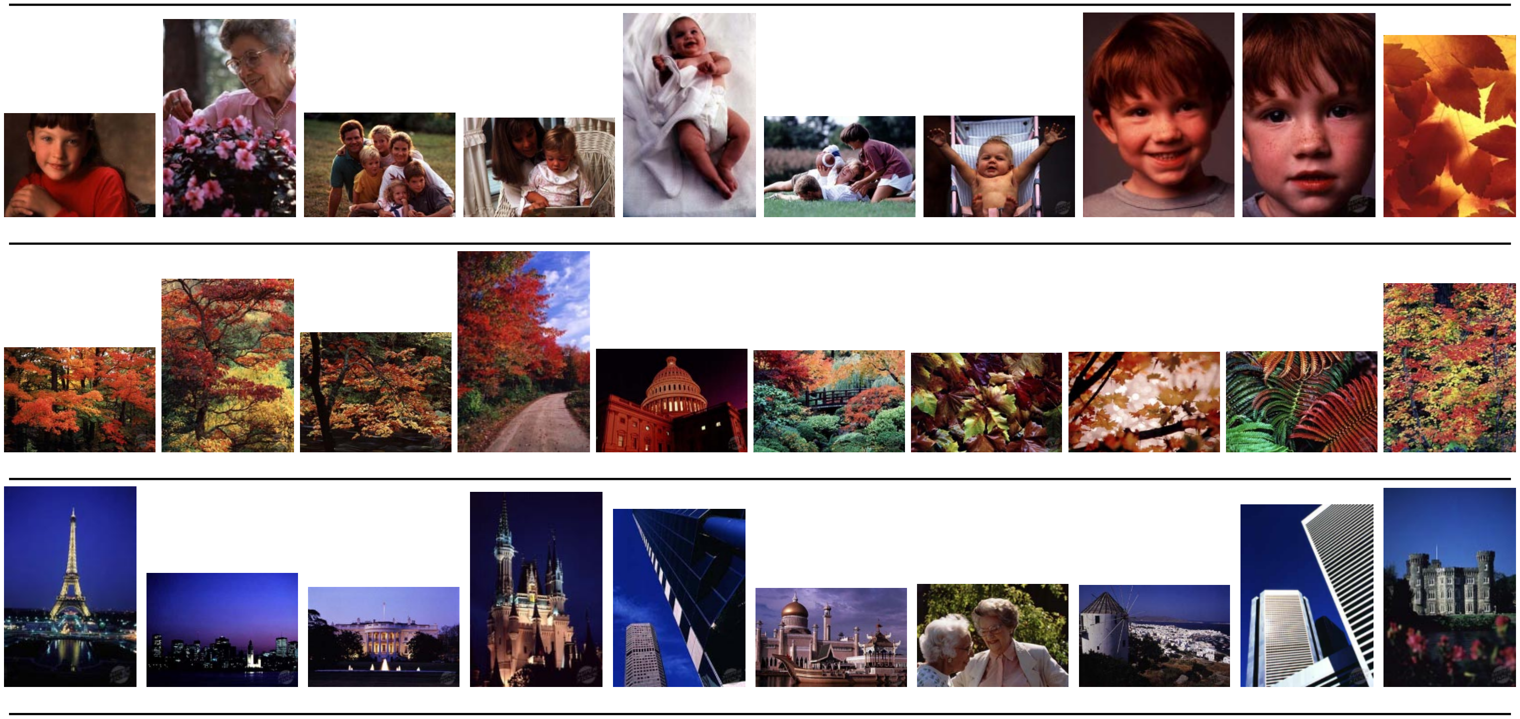


Figure 5: Three different image retrieval examples are shown (people, leaves, and sky from top to bottom). The first image on the left in each example is the image from which the query vector originates. Other spatially close vectors in the query image are then used in the multi-feedback process for ranking refinement.

4.2 Exclusive Feature Performance

To monitor the importance of the features, we made queries first using only the colour features and then using only the colour Gabor texture features. In each case the results were almost always not as good as when using the combined feature query. Hence, in the next figure we show a more interesting result. The top half of Figure 6 shows the top 10 ranked results, from left to right, after a colour only query. The subject matter is "leaves" and the system has returned 6 correct images. In the lower half of Figure 6 we can see the same query but using our colour Gabor texture features only. This time the system has retrieved 9 correct images which matches the number retrieved by our fused colour and colour texture query (compare with Figure 5). However, the fused query has achieved a better ranking of the selected images both by their subjective similarity, e.g. the building image is ranked lower by 2 places, and by their colour similarity, although this may not be distinguishable or important to the human observer across the overall results.

4.3 Performance of the Multi-Feedback Process

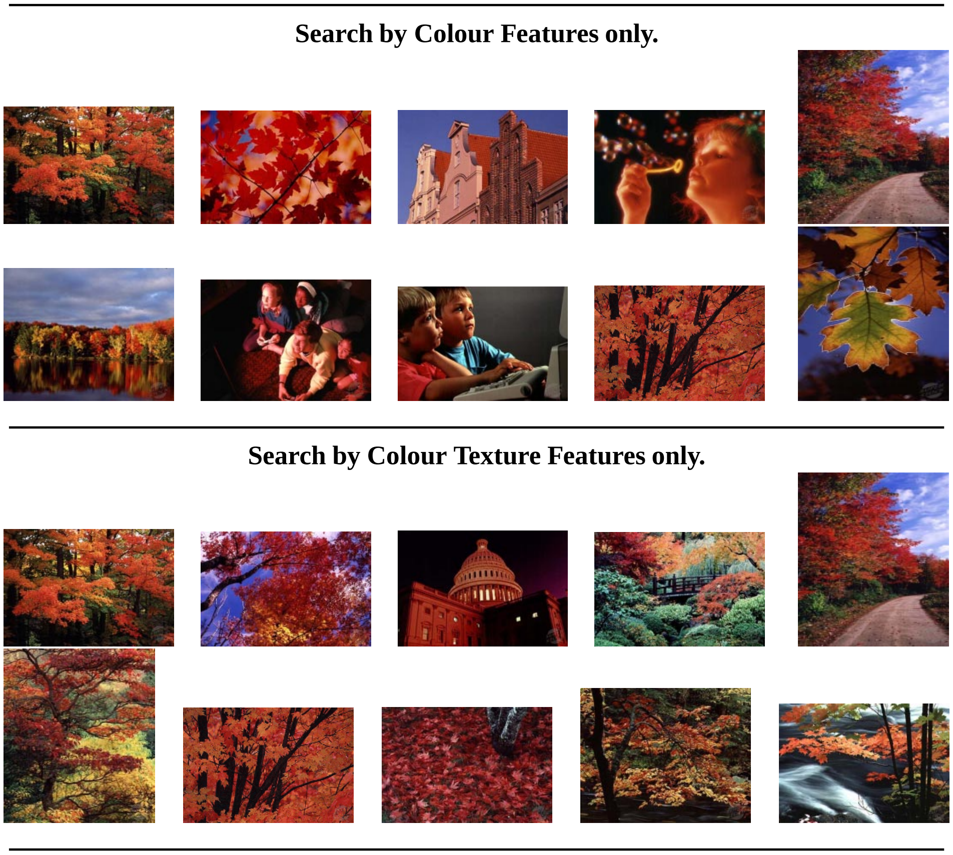
The proposed multi-feedback subset retrieval and ranking process has three levels in our implementation with 30 images retrieved at the first level. Of these, 20 are selected at the second level, and at the third level, the best 10 of the 20 are returned as the final answer. Table 1 shows the number of correct retrievals and the progressive refinement and ranking of the results in the top 10 at each level of feedback, i.e. out of 30, 20, and 10 at levels 1, 2, and 3 respectively. So for example, for the first query "sky", we obtained 6 correct answers in the top 10 at level 1, 8 correct answers at level 2 when a secondary region was used, and 9 correct answers at level 3 when a third common region was deployed. Now consider the results in level 1 only, with the first 10 out of 30 of retrieved images. This is equivalent to having the multi-feedback process switched off. The results vindicate the importance of feedback and refinement. The retrieval performance of our simple 20-component feature vector, including the multi-feedback refinement, on our database on a SUN Ultra 10 is in near real-time.

The vector-based approach easily facilitates the mechanism for weighting the colour and texture features to influence the search results. Each vector component in our system is weighted where the weights by default are set to 1.0 but can be user-adjusted to vary between 0.0 and 10.0. However, there are only two weight definitions: the chromaticity weighting, w_{uv}, applied to both u and v, and the Gabor weighting, w_g, which is applied uniformly to all Gabor features. This allows the user to weight the search towards colour or texture as appropriate, in the same way that we demonstrated the power of each perceptual feature set earlier when we alternatively set each weight to 0.0 and 1.0 (i.e. Figure 6).

Multi-feedback Ranking and Retrieval

Query	Sky	Leaves	Building	Tiger	Grass	People
Level 1	7	9	8	6	7	7
Level 2	8	9	9	7	8	9
Level 3	9	9	9	7	8	9

Table 1: This table shows the number of correct results in the top 10 retrievals at each level of the multi-feedback subset retrieval and ranking process.



Search by Colour Features only.

Search by Colour Texture Features only.

Figure 6: **Top** - Image retrieval using colour features only. **Bottom** - Image retrieval using colour texture features only. The top-left image in each half is also the query.

5 Conclusions

A method of deriving colour and colour texture features for image indexing and retrieval based on perceptual regions in the image was presented. Furthermore, a multi-feedback subset retrieval and ranking mechanism was designed to improve the performance and similarity retrievals. This allowed images which had more common regions to the query image to be found and ranked accordingly. Unlike most other CBIR techniques, we did not use colour histograms although this is quite easily possible since we too deal with identified regions, even though they are perceptual regions. One limitation of the current approach is the lack of shape information. We have not carefully examined how much shape information there is in the pre-segmentation perceptual regions, but it is clear that after segmentation and core clustering there is too much deformation to allow the derivation of shape features. Our emphasis instead has been on an alternative and novel way of deriving search features based on the perceptual observation of images by the human visual system.

References

[1] W. Niblack, X. Zhu, J. L. Hafner, T. Breuel, D. Ponceleón, D. Petkovic, M. Flickner, E. Upfal, S. I. Nin, S. Sull, B. Dom, B.-L. Yeo, S. Srinivasan, D. Zivkovic, and M. Penner, "Updates to the QBIC system," in *Storage and Retrieval for Image and Video Databases VI*, vol. 3312, pp. 150–161, SPIE, 1998.

[2] J. R. Smith and S.-F. Chang, "Querying by color regions using the VisualSEEk content-based visual query system," in *Intelligent Multimedia Information Retrieval* (M. T. Maybury, ed.), AAAI Press/MIT Press, 1997.

[3] A. Pentland, R. Picard, and S. Sclaroff, "Photobook: tools for content-base manipulation of image databases," in *Storage and Retrieval for Image and Video Databases II*, vol. 2185, (San Jose, Califronia, USA), pp. 34–47, SPIE, February 1994.

[4] J. Eakins and M. Graham, "Content-based image retrieval," tech. rep., JISC Technology Applications Program, University of Northumbria, http://www.unn.ac.uk/iidr/report.html, 1999.

[5] M. Demarsicoi, L. Cinque, and S. Levialdi, "Indexing pictorial documents by their content: A survey of current techniques," *Image and Vision Computing*, vol. 15(2), pp. 119–141, 1997.

[6] B. Rogowitz, T. Frese, J. Smith, Bouman, and E. Kalin, "Perceptual image similarity experiments," in *Human Vision and Electronic Imaging*, vol. 3299, pp. 576–590, SPIE, 1998.

[7] B. Berlin and P. Kay, *Basic Color Terms: their Universality and Evolution*. University of California Press, 1969.

[8] M. Seaborn, L. Hepplewhite, and T. Stonham, "Pisaro: Perceptual Colour and Texture Queries using Stackable Mosaics," in *IEEE ICMCS*, vol. 1, pp. 171–176, 1999.

[9] J. de Bonet and P. Viola, "Structure driven image database retrieval," in *Neural Information Processing Systems*, 1997.

[10] B. Wandell and X. Zhang, "SCIELAB: a metric to predict the discriminability of colored patterns," in *9th Workshop on Image & Multidimensional Signal Processing*, pp. 11–12, 1996.

[11] M. Mirmehdi and M. Petrou, "Segmentation of color textures," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 2, pp. 142–159, 2000.

[12] A. Jain and F. Farrokhnia, "Unsupervised texture segmentation using gabor filters," *Pattern Recognition*, vol. 24, no. 12, pp. 1167–1186, 1991.

[13] C. Setchell and N. Campbell, "Using colour gabor texture features for scene understanding," in *7th. International Conference on Image Processing and its Applications*, pp. 372–376, 1999.

[14] A. Bovik, M. Clark, and W. Geisler, "Multichannel texture analysis using localized spatial filters," *IEEE PAMI*, vol. 12, no. 1, pp. 55–73, 1990.