Machine Vision and
Applications
© Springer-Verlag 1998

Image retrieval using efficient local-area matching

V.V. Vinod*, Hiroshi Murase

NTT Basic Research Labs, 3-1 Morinosato Wakamiya, Atsugi-shi, Kanagawa 243-01, Japan; e-mail: vinod@krdl.org.sg; murase@eye.brl.ntt.co.jp

Received: 20 January 1997 / Accepted: 5 January 1998

Abstract. We present an efficient and accurate method for retrieving images based on color similarity with a given query image or histogram. The method matches the query against parts of the image using histogram intersection. Efficient searching for the best matching subimage is done by pruning the set of subimages using upper bound estimates. The method is fast, has high precision and recall and also allows queries based on the positions of one or more objects in the database image. Experimental results showing the efficiency of the proposed search method, and high precision and recall of retrieval are presented.

Key words: Image retrieval – Color matching – Efficient search – Upper bound pruning – Precision and recall

1 Introduction

Content-based image retrieval is the task of retrieving stored images whose contents resemble a given target or query. Several features such as color, shape, texture, etc. are employed for detecting the presence of the query image in a database image [2, 3, 4, 17]. Color constitutes a powerful visual cue and is one of the features more commonly employed. Often the database is filtered using color similarity before applying other expensive feature-matching techniques. It is important that retrieval based on color similarity be fast and accurate. The commonly used measures for evaluating color similarity are color histogram intersection [13], weighted distance between color histograms [5], average color distance [10] and color adjacency information [1, 11].

Most of the proposed systems, which evaluate color similarity using histograms, compare the histogram of the query image and that of the database images. These methods base their retrieval on the color distribution of the whole image and ignore the 'contents' (objects of interest) in the image. Such an approach fails when the query image constitutes a relatively small portion of the database image. In such situations, the background pixels dominate the histogram of the database image, and it will have little or no similarity with the query image histogram. This difficulty may be overcome by matching the query image against parts of the database image. We present focused color intersection [16], which evaluates the histogram intersection between the query image and parts of the database image. By this process, image retrieval can be done based on the color similarity of contents of the image rather than the 'color content' of the image. Consequently, focused color intersection achieves higher precision and recall than matching the histogram of the whole database image against that of the query image.

Matching the query image against parts of the database image could potentially require a large number of similarity evaluations. This arises primarily from the different combinations of positions and sizes at which the query image may be present in the database images. For efficient image retrieval, it is necessary to reduce the number of similarity evaluations performed, without affecting the accuracy. The coarse-to-fine approach and exploiting the probability distribution of the features have been suggested for speeding up general multiresolution template matching [8, 9, 12, 14]. Such strategies reduce the computations to some extent, but do not guarantee the optimal solution. Moreover, for exploiting the probability distribution of features, knowledge of the distributions and an easily computable feature for the first stage is required. In general, such information will not be available a priori. We propose active search which detects the best matching subimage without searching all the possible sizes and positions. Active search is directed by upper bound estimates on the similarity measure and concentrates its efforts only on areas which have high similarity with the query image. Consequently, it provides computational efficiency without sacrificing accuracy of the results. Thus, a combination of focused color intersection and active search constitutes a fast and accurate method for retrieving or filtering database images based on color similarity.

The contributions of this paper are (i) focused color intersection for matching the query image against parts of the database image providing higher retrieval accuracy and

* V.V. Vinod is currently with Kent Ridge Digital Labs, Singapore 119613

Correspondence to: H. Murase

(ii) active search for efficiently matching the query image against the large number of potential regions in a database image. In addition to higher accuracy and efficiency, the method also provides the position and approximate size of the query image in the retrieved/filtered database images. These positions may be used for resolving queries based on spatial arrangements. Also, expensive matching techniques, such as deformable templates [2, 6], texture matching [4] and spatial distribution matching [15] can be confined to the detected position of the query image in the filtered database images for saving computations. Thus, focused color intersection with active search presents an efficient and accurate method for retrieving and filtering database images based on the color distribution of its contents.

Focused color intersection is briefly discussed in Sect. 2. Active search is presented in Sect. 3. Experimental results, demonstrating the higher accuracy and efficiency of retrieval and filtering, are given in Sect. 4. The conclusions are presented in Sect. 5.

2 Focused color intersection

Focused color intersection matches the query image against parts of the database images, taking into account all positions and sizes. Details of focused color intersection may be found in [16]. Below we give salient aspects of focused color intersection, relevant for image retrieval.

The content of interest, represented by the query image, may occur at different positions and sizes in different database images. In the ideal situation, the query image should be compared against that part of the database image which exactly contains its instance. However, without a priori information regarding the objects shape, orientation and deformations, it would not be possible to ensure this in the general case. Considering all possible shapes and their orientations would be clearly infeasible. Therefore, we propose to focus on different parts of the image of some fixed shape, taking into account different sizes and positions and to match the query image against these parts. Such parts of the database image are called focus regions. In the absence of a priori information, any shape chosen for the focus region would be as good as any other. For clarity of presentation, we consider square-shaped focus regions of $w \times w$ pixels. Extensions to other regular shapes are straightforward. Arbitrary shapes may be accommodated by masking pixels in regular shapes and considering only unmasked pixels for similarity computations.

The focus regions in a database image are derived by a process of scanning the image at multiple resolutions (image size= $w, w+\Delta k, w+2\Delta k \ldots, N$). For brevity, the database image is also assumed to be square. Generalization to the more common rectangular shapes are straightforward. At each resolution, the image is scanned by shifting a $w \times w$ window by s pixels along one direction at a time. A focus region R^k_{ij} obtained from the image resized to $k \times k$ pixel may be characterized as follows.

$$R^k_{ij} = \mathbf{p}^k_{xy} \quad x = i, \ldots i+w-1, \quad y = j, \ldots j+w-1,$$
$$\mathbf{p}^k_{xy} = \mathbf{p}_{uv}, \; u = \left\lfloor \frac{xN}{k} \right\rfloor v = \left\lfloor \frac{yN}{k} \right\rfloor, \qquad (1)$$

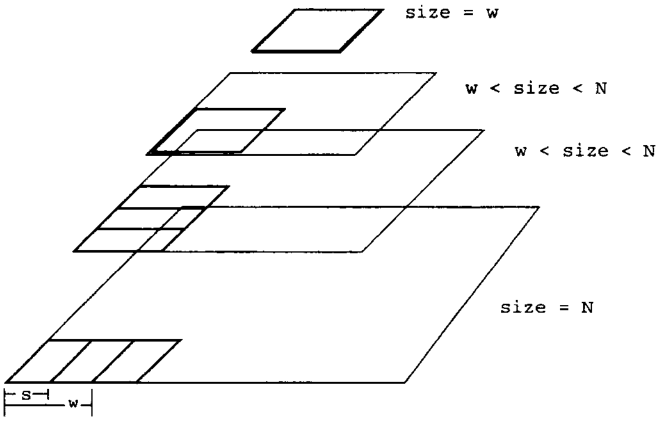


Fig. 1. The pyramid constituted by focus regions from images at multiple resolutions

where $\mathbf{p}_{xy}$ denotes a pixel in the original image and $\mathbf{p}^k_{xy}$ denotes a pixel in the image resized to $k \times k$ pixels. The whole set of focus regions are given by:

$$\text{focus regions} = \{R^k_{ij}\} \quad k = w, w+\Delta k, \ldots, N$$
$$\text{and } i = 1, s, 2s, \ldots, k; \quad j = 1, s, 2s, \ldots, k.$$

The set of all focus regions derived from a given image constitutes a pyramid. This pyramid is schematically shown in Fig. 1. At its apex, the pyramid has a single focus region covering the whole of the database image. At the base of the pyramid, the image size is the largest and the focus region represents the smallest areas matched against the query image. The size of the largest region and the smallest region matched against the query can be controlled by choosing an appropriate range of image sizes. The density of the focus regions depend on the choice of s. A value of $s = 1$ denotes the highest density, and larger values give more sparsely placed focus regions.

The similarity between a focus region and the query image is evaluated by the histogram intersection [13] between the normalized color histogram of the focus region and that of the query image. The normalized histograms are constructed by dividing each histogram cell by the total number of pixels histogrammed. Let h^M denote the normalized histogram of the query image and h^R denote the histogram of focus region R^k_{ij}. Then, the similarity S^k_{ij} between the focus region and query image is defined as

$$S^k_{ij} = \sum_{i=1}^{b} \min(h^M_i, h^R_i),$$

where h^M_i, h^R_i denote the value of the i^{th} histogram cell and b is the total number of cells in the histogram. It may be mentioned that, for evaluating S^k_{ij}, it is not necessary to compute the normalized histogram of every focus region. S^k_{ij} can be evaluated using w^2 initializations, w^2 comparisons and w^2 additions (all integer operations) from an indexed representation of the image [16]. If the highest similarity exceeds a given threshold θ, then the corresponding focus region is assumed to constitute the query image. These images are retrieved and ranked according to the similarity values. The center of the focus region which has the highest similarity with the query gives the position of the query in the database image.

3 Active search for query image

In order to retrieve all database images, containing the query, we have to search for the query image in each database image. Considering all positions and sizes ranging from w to N, there will be a total of $\sum_{n=w}^{N}(n-w+1)^2$ focus regions in each image. In practice, it may be sufficient to consider a limited set of sizes and scan the image more coarsely. Also, if the query is a key on which the database is indexed, some images may be pruned. Even then, the number of focus regions to be searched will be quite large. A naive approach will be to match the query image against all the focus regions. This operation will be computationally very expensive. For fast and accurate image retrieval, an efficient strategy for reducing the computational effort without sacrificing accuracy will be required. It may be observed that neighboring focus regions in an image are highly correlated. This suggests that their similarities will also be correlated. We exploit this fact for efficiently searching the pyramid for the best matching focus region, without actually matching all the focus regions.

Consider a focus region with low similarity with the query image. The focus regions in its neighborhood have a large number of pixels in common with it. Therefore, these regions also will have a low similarity. On the other hand, a high similarity focus region will have regions with high similarity in its neighborhood. The difference between the similarity of two neighboring focus regions can arise only from the pixels which are in one and not in the other. The number of pixels by which two regions differ can be computed based on the shape of the focus regions and their relative positions. Thus, given the similarity of a focus region, we can estimate an upper bound on the similarity of a focus region in its neighborhood. The similarity of the second region needs to be evaluated only if the upper bound is sufficiently high. Active search efficiently exploits this property. It employs upper bounds on the similarity measure for directing the search towards promising focus regions in the image. The upper bound is derived below considering two focus regions A and B, as shown in Fig. 2.

Theorem. *Given two focus regions A and B, with similarity S_A and S_B and $|A| \geq |B|$,*

$$S_B \leq \frac{\min(S_A|A|, |A \cap B|) + |B - A|}{|B|},$$

where $|A|$, $|B|$, $|A \cap B|$ and $|B - A|$ denote the number of pixels in the respective regions as shown in Fig. 2.

Proof. Let h^M, h^A and h^B denote the normalized histograms of the query image, and the focus regions. Let H^A and H^B denote the unnormalized histograms of A and B. Then,

$$S_B = \sum_i \min(h_i^M, h_i^B) = \frac{\sum_i \min(|B|h_i^M, H_i^B)}{|B|}.$$

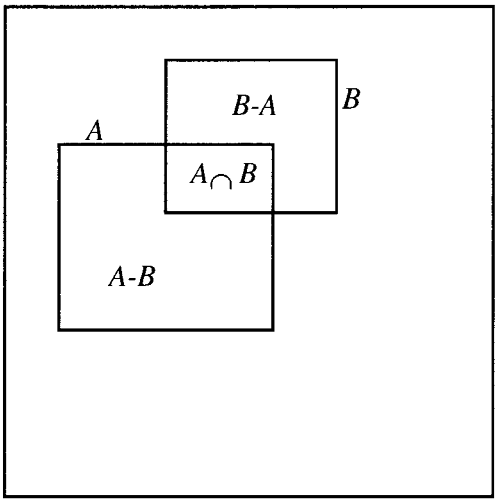


Fig. 2. Two intersection focus regions A and B

Now, $H_i^B = (A \cap B)_i + (B-A)_i$, where $(A \cap B)_i$ and $(B-A)_i$ denote the the number of pixels mapping to histogram cell i from the respective regions. We may write

$$\begin{aligned} |B|S_B &= \sum_i (|B|h_i^M, (A \cap B)_i) + (B-A)_i) \\ &\leq \sum_i (|B|h_i^M, (A \cap B)_i) + \sum_i (B-A)_i \\ &\leq \sum_i (|A|h_i^M, (A \cap B)_i) + |B - A|. \end{aligned}$$

Now, $\sum_i(|A|h_i^M, (A \cap B)_i) \leq \sum_i(|A|h_i^M, A_i) = |A|S_A$ and $\sum_i(|A|h_i^M, (A \cap B)_i) \leq \sum_i(A \cap B)_i = |A \cap B|$. Therefore,

$$S_B \leq \frac{\min(S_A|A|, |A \cap B|) + |B - A|}{|B|}.$$

The above result holds for any shape of the focus regions and the only constraint is that $|A| \geq |B|$. For focus regions with parameterized shapes, computation of $|A \cap B|$ and $|B - A|$ will be straightforward and easy. For arbitrary shapes, these quantities may have to be precomputed. The upper bound for the square focus regions defined in Sect. 2 can be computed as follows.

Consider two focus regions R_{ij}^k and R_{uv}^p such that $k \leq p$. Let S_{ij}^k denote the similarity of R_{ij}^k and let $\hat{S}_{uv}^p$ denote an upper bound on the similarity of R_{uv}^p. In order to estimate $\hat{S}_{uv}^p$, we first project both the regions on to a common image size $N \times N$. Let R_{kij}^N and R_{puv}^N denote the respective projections. Since $k \leq p$, the condition $|R_{kij}^N| \geq |R_{puv}^N|$ is satisfied. Let w_k denote the size of R_{kij}^N and (i_k, j_k) its bottom left corner. Then,

$$w_k = \frac{wN}{k} \quad i_k = \frac{iN}{k} \quad j_k = \frac{jN}{k}.$$

Similarly, for R_{puv}^N, $\quad w_p = \frac{wN}{p}, \quad u_p = \frac{uN}{p}, \quad v_p = \frac{vN}{p}.$
Ignoring the sampling effects, we obtain the upper bound as

$$\hat{S}_{uv}^p = \frac{\min(|R_{kij}^N \cap R_{puv}^N|, \alpha_{ij}^k |R_{kij}^N|) + |R_{puv}^N - R_{kij}^N|}{|R_{puv}^N|}.$$

Now, $|R^N_{puv}| = w_p^2$, and $|R^N_{puv} - R^N_{kij}| = |R^N_{puv}| - |R^N_{puv} \cap R^N_{kij}|$. And $|R^N_{puv} - R^N_{kij}$ is obtained as

$$|R^N_{puv} \cap R^N_{kij}| = 0 \quad \text{if } x_r \leq x_l \text{ or } y_t \leq y_b,$$
$$= (x_r - x_l)(y_t - y_b) \quad \text{otherwise},$$

where $x_l = \max(i_k, u_p)$, $x_r = \min(i_k + w_k, u_p + w_p)$ and $y_l = \max(j_k, v_p)$, $y_r = \min(j_k + w_k, v_p + w_p)$

Active search starts at the apex of the pyramid and progresses towards the base of the pyramid. This ensures that a focus region encountered later in the search covers lesser or equal areas as those encountered earlier, thereby satisfying the condition of the theorem. At each level of the pyramid, the search proceeds from left to right and bottom to top. A focus region's similarity is evaluated only if the least among its upper bounds exceeds the threshold and the highest similarity encountered so far. After evaluating the similarity of a focus region, upper bounds are estimated for other regions in its neighborhood. By this process, active search concentrates its efforts on focus regions with high similarity. Since the search is purely driven by upper bound estimates, it is clear that the results will be the same as exhaustively searching all the regions. Thus, active search achieves computational efficiency without sacrificing accuracy. The active search algorithm which detects the best matching focus region with similarity greater than the threshold θ is given below.

Algorithm

```
1 Set $\theta' = \theta$, $k = w$, $i = j = 1$ and $e^p_{uv} = 1.0$ for $p = w, \ldots, N$, $u = 1, \ldots, p - w + 1$, $v = 1, \ldots, p - w + 1$.
2 If $e^k_{ij} < \theta'$, set $S^k_{ij} = 0$ and go to step 5.
3 Evaluate $S^k_{ij}$. If $S^k_{ij} > \theta'$, then $R_{\text{best}} = R^k_{ij}$ and $\theta' = S^k_{ij}$.
  endif
4 Compute $\hat{S}^p_{uv}$ for all $R^p_{uv}$ in the neighborhood of $R^k_{ij}$.
  Set $e^p_{uv} = \min(e^p_{uv}, \hat{S}^p_{uv})$. The neighborhood is defined by $p \geq k$ and ($u > i$ or $v > j$), and only focus regions in the boundary have upper bounds $\geq 1$.
5 Set $i = i + 1$. If $i > k - w + 1$, then set $i = 1$, $j = j + 1$.
  If $j > k - w + 1$, then set $j = 1$, $k = k + 1$.
6 If $k \leq N$, then goto step 2.
7 If $\theta' > \theta$, then $R_{\text{best}}$ is the best matching region with similarity $\theta'$.
```

4 Experimental results

The proposed method was employed for color-similarity-based image retrieval from databases of different types of images, such as professional photographs, laboratory scenes containing common objects, animation frames, movie frames, images of sports events, scanned paintings, etc. In this section, sample results are presented and compared against other color-similarity-based retrieval.

The efficiency of active search in image retrieval and filtering and the accuracy of local color matching is presented with example results. In order to demonstrate the efficiency of active search, the average number of similarity evaluations are presented. The higher accuracy of local color matching is demonstrated by comparing against whole-image color matching. It may be mentioned that typical image retrieval systems [3, 17] consider several features for matching and some consider spatial layout of different query images in addition to local color similarity. Since the objective is to highlight the advantage of the proposed local-color-matching method, the results are compared against whole-image histogram matching. The query images were constituted by images of objects grabbed/scanned separately or clipped from a database image. A human observer inspected all the database images for the query images and the database images were ranked according to visual similarity of contents with the query image. The results of the algorithms (local color matching and whole-image histogram intersection) are compared with respect to this ranking by a human observer. Detailed results for the following three datasets are discussed below.

1. Images: 60 scenes of a laboratory table with common objects against complex backgrounds. Models: images of single objects grabbed separately. Sample image and model is shown in Fig. 3.
2. Images: 100 images of teddy bears including indoor and outdoor scenes. Queries: images clipped from some scenes. Sample image and model shown in Fig. 4.
3. Images: 150 frames of an animation movie. Query: one character clipped from a frame.

The RGB histogram with 16 divisions along each axis was used in all the experiments. It may be mentioned that other color spaces and non-uniform quantizations may also be employed. The algorithm remains the same, except for the computation of the histogram and the indexed representation of the image from which the similarity is evaluated.

The computational efficiency of active search is presented in Sect. 4.1. The retrieval accuracy of focused color intersection is compared against histogram intersection of whole images in Sect. 4.2.

4.1 Efficiency of active search

Here, we present results on the efficiency of active search for image retrieval. The number of focus regions in an image depends on the image size, window size, etc. For studying the efficiency of active search, all the images which are originally of different sizes are scaled to 128 × 128 pixels. A 32 × 32 window was used for scanning the images. The window was shifted by one pixel at a time (i.e., $s = 1$) and all image sizes from $w \times w$ to 128×128 were included in the pyramid. This results in a total of 308,945 focus regions. The number of focus regions out of this total which are matched by active search is studied.

Typically, color similarity is employed for two different but related tasks in content-based image retrieval systems. The efficiency of active search is studied with respect to each of these. In one, the task is to divide the database images into two classes – one which may contain the image and the other which does not contain the image. Typically, all the images which may contain the query image are further verified using other features. In this case, it is sufficient to determine if there is a enough evidence based on color similarity for the query image to be present in a database

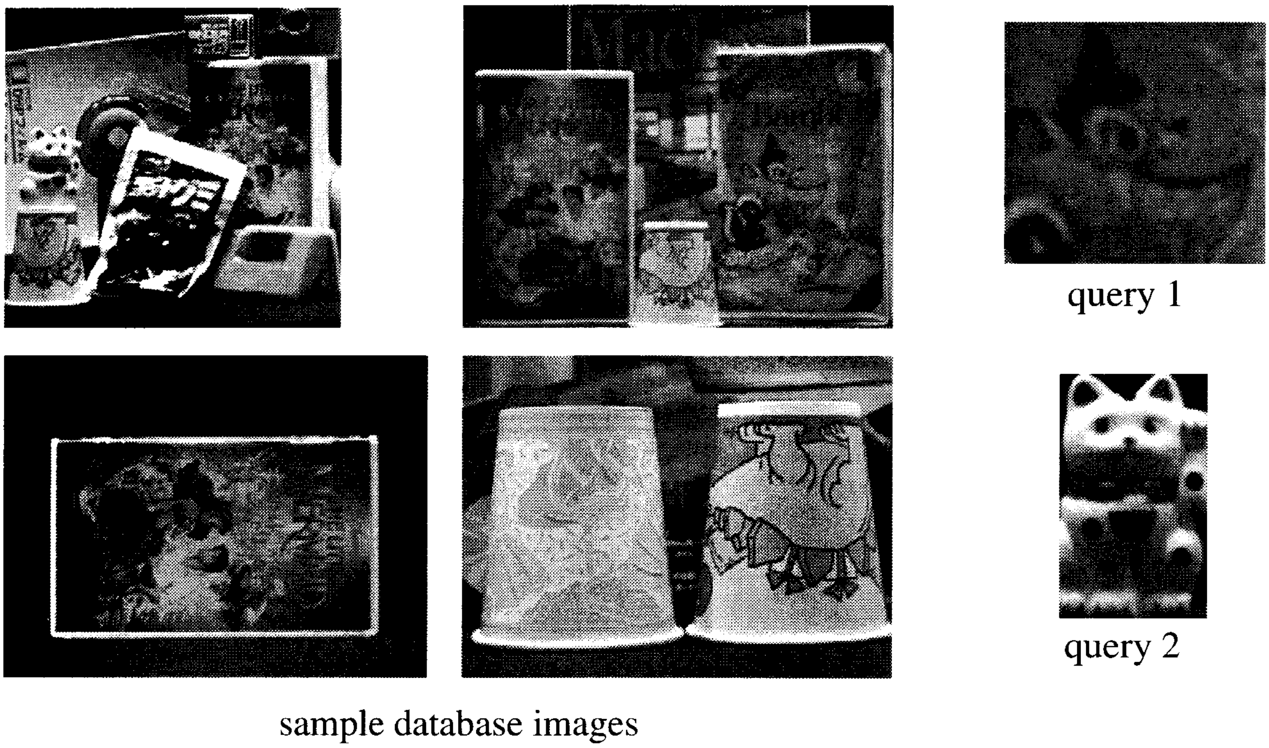


Fig. 3. Sample database images and query images for the database of laboratory scenes

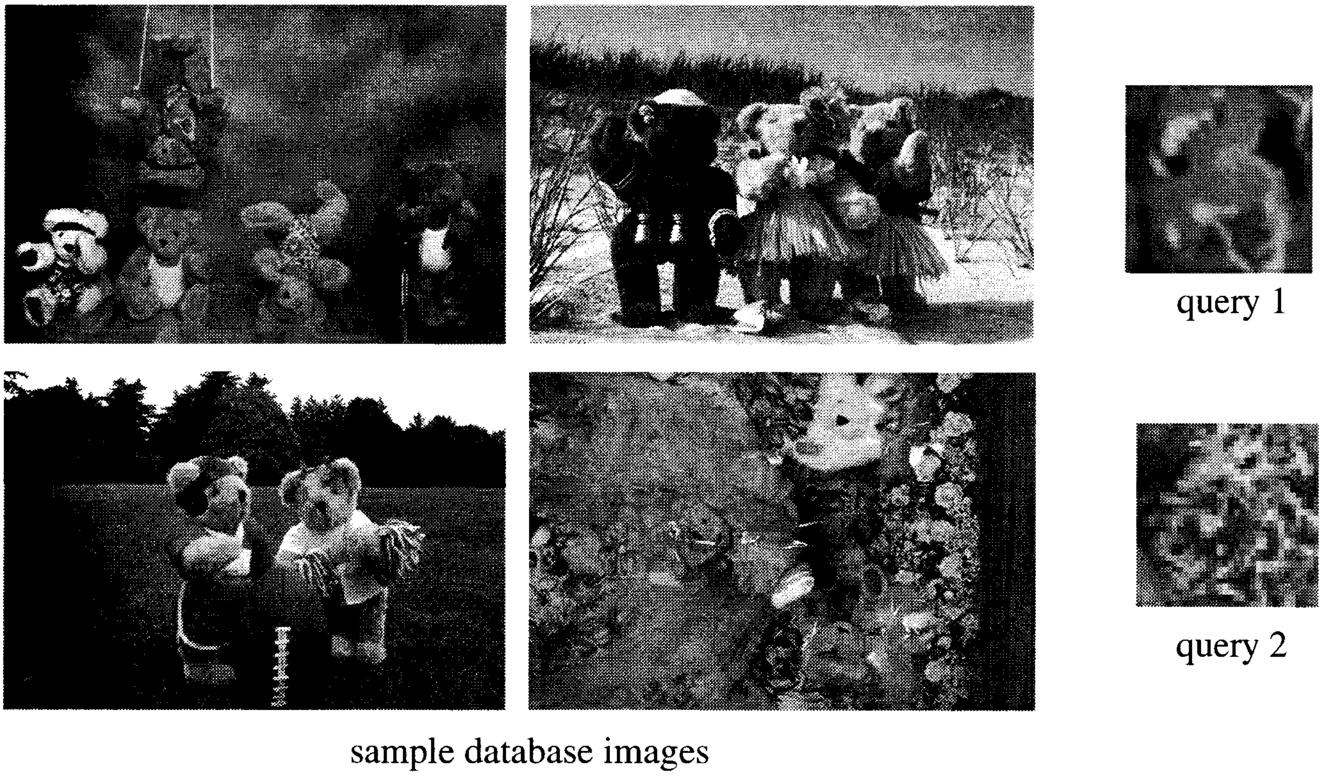


Fig. 4. Sample database images and query images for the database of teddy bear images

image. The filtered images are not ranked by the strength of the evidence. This operation is typically required when the aim is to retrieve all images containing a given query. In this case, active search for the query image can be terminated as soon as the first focus region with similarity above a given threshold is encountered. We term this case as the *first match*, since the search can be terminated as soon as the first matching focus region is encountered.

The other type of color-similarity-based retrieval is required when the user desires to retrieve the best n matching images from the database. In this case, it is not merely sufficient to determine the images which may contain the query image, but they have to be ranked also. Hence, active search will have to determine the focus region which has the highest similarity with the query image and cannot be terminated as soon as a focus region with similarity exceeding the threshold is encountered. Ranking can then be done based on the highest similarities of the database images against the query image. Additional feature matching may be applied to the ranked images in order of their ranking till the desired number of database images are retrieved. We term this case as the *best match*, since active search for the query image has to determine the best matching focus region in each database image.

Experiments were conducted separately with each database. The similarity threshold was fixed at 0.3. In the best match case, the task was to retrieve and rank all images which had a focus region whose similarity with the query exceeded 0.3. Fourteen queries were posed to the first database, six to the second and one to the third. For each query, the fraction of the database retrieved (F_R) and the average number of focus regions matched (N_{avg}) were computed as follows.

$$F_R = \frac{\text{no. of images retrieved}}{\text{no. of images in the data set}},$$

$$N_{\text{avg}} = \frac{\text{no. of histogram intersection evaluations}}{\text{no. of images in the data set}}.$$

In the first match case, active search for an image was terminated as soon as a focus region with similarity above 0.3 was encountered. For each query, F_R and N_{avg} are computed as in the case of best match. The results for best match and first match are shown in Fig. 5.

The results show that active search examines only a very small fraction of the total number of focus regions. The maximum number of focus regions matched in the best match case is less than 1% of the total number of 308,945 regions. For the first match case, the number of focus regions matched is still less. These results demonstrate the efficiency with which active search directs the search towards promising focus regions in the image. Active search results in a large computational gain. Since the search is directed by upper bounds, the results will be the same as performing an exhaustive search. Thus, active search achieves large reduction in computational effort without sacrificing accuracy.

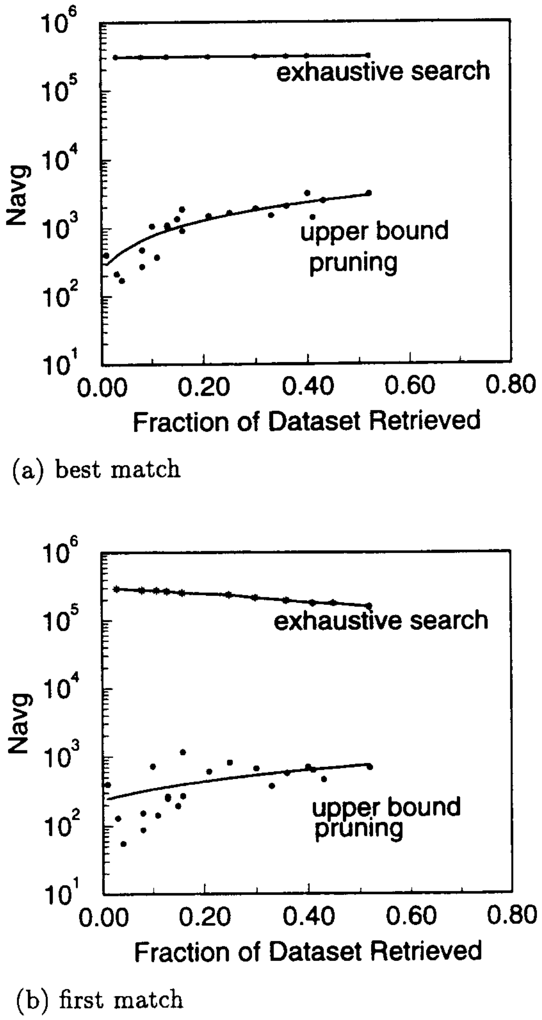


Fig. 5a,b. Average number of focus regions matched per image against fraction of images retrieved for best match and first match

4.2 Retrieval accuracy of focused color intersection

In this section, we discuss the retrieval accuracy of focused color intersection. Focused color intersection is compared against histogram intersection of the whole image. The comparison is done on the basis of precision and recall for each query.

Let n be the number of database images which contain the query image. Let n_r denote the number of images retrieved, of which n_c are correct (i.e., they contain the query image). Then, precision and recall are given by

$$\text{precision} = \frac{n_c}{n_r}, \qquad \text{recall} = \frac{n_c}{n}.$$

Now, precision provides a measure of the false retrievals and recall provides a measure of the misses. Higher precision denotes less false retrievals and higher recall indicates less misses. Perfect precision (= 1.0) denotes no false retrievals and perfect recall (= 1.0) denotes no misses. Precision and recall also indicate the effectiveness of ranking the retrieved images. A high precision and high recall indicates that most of the correct images are ranked high. If any one of the values is low, it indicates that a large number of high-ranked images are false retrievals. If both values are low, then most of the high-ranked images are false retrievals. It is clear that either measure alone does not indicate the accuracy of a retrieval. Both precision and recall should be high and the best case is achieved when both equal 1.0. We compare

Table 1. Comparison of precision and recall for database of laboratory scenes

Query image 1

Focused color intersection		Whole-image intersection	
precision	recall	precision	recall
1.00	1.00	1.00	0.72
1.00	1.00	0.91	1.00

Query image 2

Focused color intersection		Whole-image intersection	
precision	recall	precision	recall
1.00	0.50	1.00	0.25
0.67	0.75	0.57	0.50
0.63	1.00	0.50	1.00

the performance based on precision and recall of focused color intersection and whole-image histogram matching. In the experiments, n and n_c involved in the calculations of precision and recall were determined by manual inspection and n_r was given by the algorithm.

Database of laboratory scenes

Some images belonging to this database and two query images are shown in Fig. 3. The database consisted of a total of 60 images of objects on a laboratory table against complex colorful backgrounds. The images were of varying sizes, ranging from 240 × 240 to 320 × 240. A human oberver determined that the first query image was present in 11 database images and the second query image was present in 8 database images.

Given the first query image, focused color intersection retrieved all the 11 correct database images without retrieving any false images. That is, the correct database images were ranked 1 to 11. This gives a precision and recall of 1.0. In the case of histogram intersection of whole images, the image ranked 8 was incorrectly retrieved. That is, to achieve perfect recall, one false retrieval was necessary. The corresponding precision and recall values are given in Table 1. For the second query image, focused color intersection retrieved 3 false images (ranked 5, 6, and 7) for achieving perfect recall. Matching whole-image histograms required 7 false retrievals (ranks 3, 4, 5, 8, 10, 11, 12), resulting in a precision of only 0.5 for perfect recall. The precision recall values may be seen in Table 1.

Database of teddy bear images

This database consists of 100 images of teddy bears in various environments, including both indoor and outdoor scenes. Sample images and two query images are shown in Fig. 4. The query images have been clipped from database images and indicate the swimsuit worn by teddy bears in some of the images. Manual observation determined that the first query image was present in 5 database images and the second query image in 4 database images.

Focused color intersection retrieved all the images containing the first query image without any false retrievals (correct images ranked 1 to 5). This gives a precision and recall of 1.0. For the same query image, histogram intersection of the whole images, did not give any correct images in the 9 top-ranked images. The database image from which the query image was clipped was ranked 10th. The other correct retrievals were ranked 21, 23, 29 and 36, which results in very low precision. In the case of the second query image, focused color intersection retrieved one false image (ranked 3) for perfect recall. The correct images were ranked 3, 11, 46 and 63 by whole-image histogram intersection. This results in very low precision for high recall and vice versa. The precision and recall values are given in Table 2.

Table 2. Comparison of precision and recall for database of teddy bear images

Query image 1

Focused Color Intersection		Whole Image Intersection	
precision	recall	precision	recall
1.00	1.00	0.10	0.20
1.00	1.00	0.09	0.40
1.00	1.00	0.13	0.60
1.00	1.00	0.14	0.80
1.00	1.00	0.14	1.00

Query image 2

Focused Color Intersection		Whole Image Intersection	
precision	recall	precision	recall
1.00	0.25	0.33	0.25
1.00	0.50	0.18	0.50
0.75	0.75	0.07	0.75
0.80	1.00	0.06	1.00

Table 3. Precision and recall of focused color intersection and whole image histogram matching for animation frames

Query image 1

Focused Color Intersection		Whole Image Intersection	
precision	recall	precision	recall
0.75	0.90	0.60	0.90
0.79	0.85	0.75	0.53

Database of animation frames

The animation frames database consisted of 150 frames of an animation movie. The query image was a character in the movie clipped from one of the frames. Manual inspection showed that 43 frames of the movie clip contained the desired character. That is, $n = 43$. The frames covered different scenes of the movie and had considerable variations in orientation of the objects, lighting, etc. The objective was to retrieve all frames containing the desired character. Focused color intersection was able to retrieve the images with high precision and recall. On the other hand, histogram intersection of whole images had considerably lower precision and recall. The precision and recall values in the range 0.4–1.0 are plotted in Fig. 6. The best values obtained by each method (closer to (1.0,1.0)) are given in Table 3.

From the precision and recall values observed for different databases and queries, it is seen that focused color intersection has consistently higher retrieval accuracy than matching whole-image histograms. While matching the histogram of whole images, the background pixels may domi-

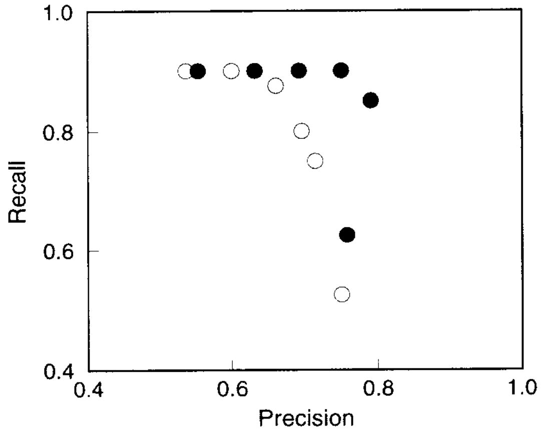


Fig. 6. Precision and recall of focused color intersection (•), histogram intersection of whole images (o) for animation frames database

nate the histogram, resulting in poor retrieval accuracy. By focusing on parts of the image, background effects are minimized and higher retrieval accuracy is obtained. In many cases, focused color intersection achieves near-perfect recall and precision. Thus, focused color intersection is ideally suited for retrieving images based on the color of its contents. Since active search matches only a few focus regions, the higher retrieval accuracy is achieved at little extra cost.

5 Conclusion

In this paper, we have proposed an efficient and accurate method for retrieving images based on the color of its contents. The method matches the query image against parts of the database image. The search for the best matching subimage is efficiently done by active search using an upper bound on the similarity measure. Active search achieves speed-up without sacrificing accuracy. Focused matching retrieves the images based on color of the contents of the image rather than on 'color content' of the image, which often represents the background rather than an interesting object in the scene. Consequently, focused color intersection with active search achieves higher retrieval accuracy.

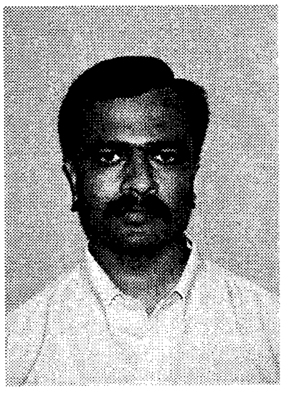
In the present work, active search employs upper bounds on the normalized histogram intersection. However, the idea of active search does not depend on the similarity measure itself, but on the availability of an easily computable upper bound. Active search using other feature similarities is being explored.

Focused color intersection also gives the position of the query in the database image, enabling position-based retrieval. Other costly feature-matching techniques may be applied at the detected position. Detecting the position also enables queries based on multiple objects in a scene. These aspects are under investigation.

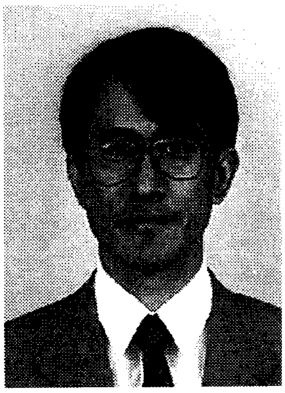
Acknowledgements. The authors wish to thank Dr. T. Izawa, Dr. K. Ishii and Dr. S. Naito and Ms. C. Hashizume of NTT Basic Research Labs for their help and encouragement in conducting this research.

References

1. Chua T-S, Lim S-K, Pung H-K (1994) Content-based retrieval of segmented images. In: Proc. Second ACM Int. Conf. on Multimedia, 1994, San Francisco, October 1994, pp 211–218
2. Del Bimbo A, Pala P (1996) Effective image retrieval using deformable templates. In: Proc. ICPR'96, Vienna, August 1996, Vol C, pp 120–124
3. Flickner M, et al. (1995) Query by image and video content: The QBIC system. IEEE Comput 28 (9): 23–32
4. Gimmelfarb GL, Jain AK (1996) On retrieving textured images from an image database. Pattern Recogn 29 (9): 1461–1483
5. Hafner J, Sawhney HS, Equitz W, Flickner M, Niblack W (1995) Efficient color histogram indexing for quadratic-form distance functions. IEEE Trans Pattern Anal Mach Intell 17 (7): 729–736
6. Jain AK, Zhong Y, Lakshmanan S (1996) Object matching using deformable templates. IEEE Trans Pattern Anal Mach Intell 18 (3): 267–278
7. Jain AK, Vailaya A (1996) Image retrieval using color and shape. Pattern Recogn 29 (8): 1233–1244
8. Margalit A, Rosenfeld A (1990) Using feature probabilities to reduce the expected computational cost of template matching. Comput Vision Graphics Image Process 52: 110–123
9. Margalit A, Rosenfeld A (1990) Using probabilistic domain knowledge to reduce the expected computational cost of template matching Comput Vision Graphics Image Process 51: 219–234
10. Mehtre BM, Kankanhalli MS, Narasimhalu AD, Man GC (1995) Colo matching for image retrieval. Pattern Recogn Lett 16: 325–331
11. Nagasaka A, Tanaka Y (1992) Automatic Video Indexing and Full-Video Search for Object Appearances. Elsevier, Amsterdam, pp 113–127
12. Rosenfeld A, Vanderburg GJ (1977) Coarse-fine template matching. IEEE Trans Syst Man Cybern 7: 104–107
13. Swain MJ, Ballard DH (1991) Color indexing. Int J Comput Vision 7 (1): 11–32
14. Vanderburg GJ, Rosenfeld A (1977) Two-stage template matching. IEEE Trans Comput C-26: 384–393
15. Vinod VV, Murase H (1996) Object location using complementary color features: histogram and DCT. In: Proc. ICPR'96, Vienna, August 1996, Vol A, pp A-554-559
16. Vinod VV, Murase H (1997) Focused color intersection with efficient searching for object extraction. Pattern recognition 30 (10): 1787–1797
17. Wu JK, Narasimhalu AD, Mehtre BM, Lam CP, Gao YJ (1995) CORE: a content-based retrieval engine for multimedia information systems. Multimedia Syst 3 (1): 25–41



V.V. Vinod received the B.Tech. (Hons), M.Tech. and Ph.D. degrees in compute science and engineering from the Indian Institute of Technology, Kharagpur, India in 1988, 1990 and 1994, respectively. He is currenlty a research staff at Kent Ridge Digital Labs, Singapore, working on visual media content management and manipulation. During 1995–1996, he was with NTT Basic Research Labs, Japan, where he worked on content-based retrieval of images and video. He has worked with the Indian Institute of Technology, Kharagpur (1989–1993), Electronics Research and Development Centre, Trivandrum (1993–1994) and Ashok Leyland Information Technology Ltd., Bangalore (1994–1995). His research interests include neural networks, genetic algorithms, pattern recognition, image processing, video analysis and visual media content management. He is a member of IEEE.



Hiroshi Murase received the B.E., M.E., and Ph.D. degrees in electrical engineering from the University of Nagoya, Japan. From 1980 to the present, he has been engaged in pattern recognition research at Nippon Telegraph and Telephone Corporation (NTT). From 1992 to 1993, he was a visiting research scientist at Columbia University, New York. He was awarded an IEICEJ Shinohara Award in 1986, and a Telecom System Award in 1992, and the IEEE CVPR best paper award in 1994, and the IEEE ICRA best video award in 1996. His research interests include computer vision, video analysis, character recognition, and image recognition. He is a member of the IEEE, IEICEJ, IPSJ.