Pergamon

Pattern Recognition, Vol. 30, No. 10, pp. 1787–1797, 1997

0031-3203/97 $17.00+.00

PII: S0031-3203(96)00192-6

FOCUSED COLOR INTERSECTION WITH EFFICIENT SEARCHING FOR OBJECT EXTRACTION

V. V. VINOD† and HIROSHI MURASE*

NTT Basic Research Labs, Morinosato Wakamiya, Atsugi-shi 243-01, Japan

(*Received* 2 *November* 1995; *in revised form* 21 *November* 1996)

Abstract—We propose focused color intersection with efficient searching for identifying and extracting the objects in a complex scene based on color similarity. The method matches the models against different parts of a scene, called focus regions, using normalized color histogram intersection. The best matching focus region is determined by an efficient search strategy employing upper bound pruning. This search strategy, called active search, concentrates its effort on parts of the scene having high similarity with the object. Consequently, it achieves a large reduction in computational effort without sacrificing accuracy. An efficient algorithm for evaluating the color histogram intersection between a model and a focus region is also given. Experiments conducted demonstrate that multiple known objects in complex scenes can be extracted by this process. The method is stable against scale changes, two-dimensional rotation, moderate changes in shape and partial occlusion. © 1997 Pattern Recognition Society. Published by Elsevier Science Ltd.

Focused matching Color intersection Competitive matching Object extraction Recognition Indexing

1. INTRODUCTION

Retrieving known objects from a complex scene is central to several practical vision applications. This task involves identifying the known objects in the scene and determining the region occupied by these objects. In addition to object recognition and scene interpretation, the applications include associative retrieval, querying image databases with visual data, search and replace operations in multimedia document editing, etc. In this paper we propose a strategy for identifying known objects and demarcating the approximate regions occupied by these objects in a complex scene using only color information.

Most of the strategies proposed for identifying and locating objects in a scene make use of geometric features.[1,2] These methods extract local features such as corners and edges from the scene and then match them against the model's local features. Extracting local geometric features of objects in a complex scene would, in general, be computationally very expensive. Changes in orientation, scaling or view point would introduce additional complexity. Also, shape-based recognition techniques cannot be applied to non-rigid objects. Template matching techniques constitute the other approach for detecting objects in a scene under small distortions and noise.[2,3] Usually several views of the model are stored and matched against the input scene to account for changes due to two- or three-dimensional orientation, scaling, shape changes and other distortions. The matching is done using moments,[2] parametric eigenspace representations,[4] etc. Irrespective of the representation scheme employed, storing and matching against several templates for the same model could be computationally very expensive. Hence it would be desirable to do the matching using features invariant to as many changes as possible. The color distribution of objects present one such feature and is adopted in this work. The color distribution is invariant to changes in two-dimensional orientation and shape. It is stable against moderate occlusion and small changes in three-dimensional orientations.

Recently, it has been demonstrated that color distributions constitute a powerful feature for image matching. Swain and Ballard[5] introduced color similarity evaluation by Histogram Intersection, for object detection and image database indexing, and Histogram Back projection for object location. It has been shown by Stricker and Swain[6] that the histogram space can be used to store sufficiently large numbers of distinguishable image histograms. Hafner *et al.*[7] proposed a class of quadratic form distance functions for evaluating color histogram similarity. Quadratic form distance functions are employed in IBM's QBIC image retrieval system.[8] Mehtre *et al.*[9] proposed two features for color matching, namely the mean values of the individual color axes and the histogram of reference colors. These features reduce the level of detail in the matching process and are applicable mainly for images with large regions of uniform color. The CORE[10] image retrieval system employs these features. A color-based image retrieval system using fuzzy matching techniques has been proposed in reference (11). The different

* Author to whom correspondence should be addressed. E-mail: murase@eye.brl.ntt.jp.

† Current address: Institute of Systems Science, National University of Singapore, 119597 Singapore.

colors are segmented and their coverage and distributions are estimated. Queries based on these features are evaluated using fuzzy techniques. Schettini[12] applied color matching along with shape matching for detecting a known object against a known background. Here shape forms the primary cue and a histogram intersection value is applied for verifying the hypothesis generated by shape matching. The above approaches indicate that color constitutes an important feature for image matching. However, all the methods except that in references (11,12) confine themselves to evaluating the similarity between two images and their roles are limited to indexing image databases given most of the query image.

The color histogram of a complex scene containing several objects will be considerably different from that of the individual objects. Consequently, evaluating the similarity between the scene histogram and model histograms will fail to reliably detect the object. In such situations, matching the model histograms against a histogram of parts of the scene results in better precision and recall.[13,14] Vinod and Murase[15] and Ennesser and Medioni[16] have shown that object locations obtained by matching local histograms are better than those obtained by histogram backprojection.[5] Ennesser and Medioni[16] compute histogram intersections with all local areas of a given size considering all positions. When size is unknown, a greedy search for the locally best size is done at each location. This approach would, in general, require a large number of histogram intersection evaluations and give a locally best match. Vinod *et al.*[14] proposed an upper bound pruning search, called active search, for detecting the globally best position and size with low computational effort. Upper bound pruning skips over uninteresting areas in the image and concentrates only on promising parts of the image matching the model. Consequently, active search also detects the absence of a model very quickly. In order to detect and extract multiple objects in a scene, detecting the best matching size and position for each object will not be sufficient. A conflict resolution strategy will be necessary to ensure that the parts of an image associated with different objects are disjoint.

In this paper we propose an efficient, iterative strategy for extracting multiple known objects from complex scenes. The model histograms are matched against parts of the image, called focus regions, which are extracted from a multiresolution structure. In each iteration, active search is used for efficiently determining the best matching focus region for each model. A competitive identification and pruning step associates a subimage with a model and prunes the image and the set of models. This step ensures that image parts associated with different objects are disjoint. Pruning the image and the set of models reduces unnecessary computation. The similarity measure used is Histogram Intersection. The proposed method can operate in situations where the background is unknown, the objects are of different sizes and under considerable occlusion and overlapping of objects. It is stable against changes in two-dimensional orientation and shape. The method, however, is influenced by changes in lighting conditions and major changes in three-dimensional pose.

In Section 2 we give details of the proposed method. Experimental results obtained under various conditions are given in Section 3. A concluding discussion is presented in Section 4.

2. FOCUSED COLOR INTERSECTION WITH ACTIVE SEARCH

The following setting is considered for developing the method. "Given a set of models $\mathsf{M} = \{M_n\}, n = 1, ..., N$, where each M_n is the color image of a known object and a scene $\mathcal{I}$ of $X \times Y$ pixels (i.e. $\mathcal{I}=\mathbf{p}_{xy}$, $x = 1, ..., X$, $y = 1, ..., Y$), identify any model objects present in the scene and extract the regions occupied by them." The scene $\mathcal{I}$ may consist of zero or more known objects against a complex unknown background. The absolute as well as relative sizes of the objects may vary from scene to scene. There could be any amount of change in two-dimensional orientation and small change in three-dimensional orientation of the objects. Objects may be partially occluded and the shape of an object may vary from scene to scene.

In this section we describe the following steps of the proposed method:

- *Extracting and Matching Focus Regions*—Extracting the set of focus regions in a multiresolution structure and an efficient algorithm for evaluating the color similarity between a focus region and a model.
- *Competitive Identification and Pruning*—The process of competitively associating a focus region with a model and pruning the sets of competing focus regions and models.
- *Active Search*—Efficient object search method using upper bound pruning for determining the best matching focus region.

This section concludes with an algorithmic specification of the proposed method.

2.1. Extracting and matching focus regions

The histogram of a scene containing multiple objects will, in general, have little or no similarity to a model histogram. In such situations, we have to consider parts of the scene for matching against the models. We refer to such parts as focus regions. Ideally, the focus regions should contain a single object. However, this will be difficult to ensure in the absence of *a priori* information regarding the object size, shape, etc. Since the objects may occur at different sizes and positions in different images, the focus regions should cover all sizes and positions. However, since the color distributions of a few pixels in the scene will not carry any effective information, regions with very few pixels should not be considered.

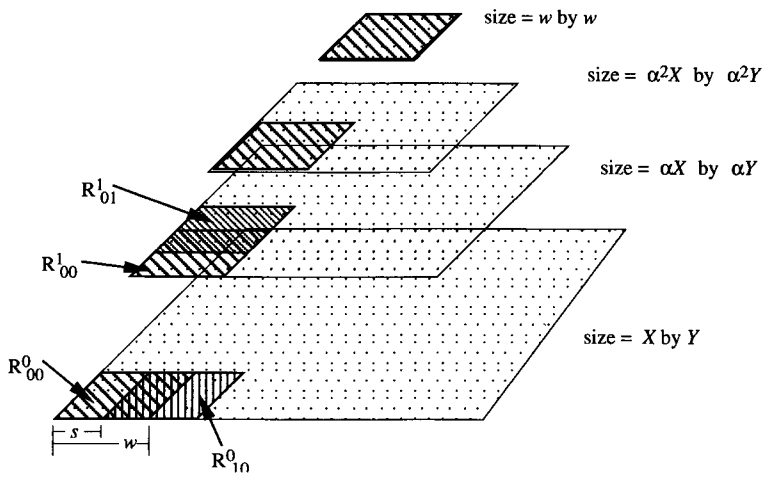

Fig. 1. The focus region extraction process. Some focus regions extracted with image resizing factor $\alpha = 0.8$ are shown.

In the absence of *a priori* information favoring any particular shape for the focus regions, a regular shape such as a circle or square may be used. For the sake of concreteness, we consider a square shape and focus regions are extracted using a square window of size $w \times w$ pixels. Different focus regions are extracted by scanning the input image with the square window. For scanning the image the window is shifted by s pixels in one direction at a time. After one complete scan is over, the input image is scaled by a factor α, where $\alpha < 1$. Focus regions are extracted from this resized image by scanning it with the same window as earlier. By this process we would be focusing upon larger regions from the original image. This will accommodate changes in object size. This process of resizing by a factor of α and scanning the input image is continued until the image becomes smaller than the scanning window. Thus, the focus regions are extracted by a process of resizing the input image by a factor α (i.e. image sizes $1,\alpha,\alpha^2,\ldots$, and scanning each resized image with a fixed size square window. Figure 1 shows this process. The hatched squares represent some of the focus regions in the images. The window size w and the shift s used for scanning the images are also shown in the figure. The set of focus regions may be characterized as follows. Let $\mathcal{I}^k$ denote the image resized by α^k, and $\mathbf{p}^k_{xy}$ denote the pixels belonging to $\mathcal{I}^k$. Then

$$\mathcal{I}^k = \mathbf{p}^k_{xy}, \quad x = 1, \ldots, \alpha^k X, \; y = 1, \ldots, \alpha^k Y,$$

where

$$\mathbf{p}^k_{xy} = \mathbf{p}_{uv}, \quad u = \lfloor \frac{x}{\alpha^k} \rfloor, \; v = \lfloor \frac{y}{\alpha^k} \rfloor.$$

Let R^k_{ij} denote a focus region belonging to $\mathcal{I}^k$. Then the set R of all focus regions considering all resized images is given by

$$\mathsf{R} = \{R^k_{ij}\}, \tag{1}$$

where

$$k = 0, \ldots, \min\left(\lfloor \log_\alpha \frac{w}{X} \rfloor, \lfloor \log_\alpha \frac{w}{Y} \rfloor\right)$$
$$i = 0, \ldots, \frac{\alpha^k X - w}{s}, \quad j = 0, \ldots, \frac{\alpha^k Y - w}{s},$$
$$R^k_{ij} = \mathbf{p}^k_{xy}, \; x = si + 1, \ldots, si + w \text{ and}$$
$$y = sj + 1, \ldots, sj + w.$$

The similarity $S(R,M)$ between a focus region R and a model M is evaluated as the histogram intersection between their normalized color histograms. The normalized color histogram is obtained by dividing each histogram count by the total number of pixels. That is, the sum of all counts in a normalized histogram will be 1.0. All references to histogram intersection in this paper shall mean histogram intersection of normalized histograms. The histogram intersection between two histograms h^M and h^R, each with b bins, is defined as:[5]

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

A straightforward computation of this measure, for a focus region and a model, would be to construct the normalized histogram of the focus region and then compute its intersection with the precomputed normalized histogram of the model. This would require initializing the entire histogram and one scan over the focus region and at least one scan over the histogram (assuming that normalized histogram intersection can be computed by one scan over the non-normalized histogram). The resulting complexity will be of order of $\max(w^2,b)$, where b is the number of bins in the histogram. However, since the focus region has only w^2 pixels, there will be at most w^2 relevant bins in the histogram. The other bins will definitely have zero values and will not contribute to the histogram intersection. Based on this observation, we use the following method for computing the normalized

histogram intersection of focus region R_{ij}^k and a model M_n.

The entire image is first converted to an internal representation by replacing the color value of each pixel by the index of the histogram bin to which that pixel is mapped. That is, each pixel $\mathbf{p}_{xy}$ in the image has an integral value indexing the histogram counts. This operation takes similar (actually less) effort as histogramming the whole image and has to be done only once. The following algorithm computes the histogram intersection from this representation and the precomputed normalized histogram of the model without explicitly constructing the histogram of R_{ij}^k.

Algorithm Evaluate

1. Focus region R_{ij}^k, Model histogram h^n, Temporary histogram h.
2. Initialize count=0, $S(R_{ij}^k, M_n) = 0$.
3. For each pixel $\mathbf{p}_{xy}^k$ of R_{ij}^k do; $h_{\mathbf{p}_{xy}^k} = 0$ if $\mathbf{p}_{xy}^k$ is not masked; count= count +1 if $\mathbf{p}_{xy}^k$ is masked
4. For each pixel $\mathbf{p}_{xy}^k$ of R_{ij}^k do; If

$$\left(h_{\mathbf{p}_{xy}^k} < h^n_{\mathbf{p}_{xy}^k}\right)$$

then

$$h_{\mathbf{p}_{xy}^k} = h_{\mathbf{p}_{xy}^k} + \frac{1}{\text{count}} S\left(R_{ij}^k, M_n\right) = S(R_{ij}^k, M_n) + \frac{1}{\text{count}}.$$

The above algorithm scans the focus region twice. In the first scan, in step 3, the temporary histogram is initialized. In the subsequent scan in, step 4, the histogram intersection is evaluated. The complexity is $O(w^2)$ and is independent of the number of histogram bins. Since complexity is independent of histogram size, the algorithm is also well suited for large histograms such as co-occurrence histograms.[16]

2.2. *Competitive identification and pruning*

In the case of a perfect match between a model M and focus region R the histogram intersection value $S(R,M)$ will be equal to 1.0. However, a perfect match is very unlikely. In general, even when R contains exactly the same object as M, the intersection value would be less than 1.0. This may be the result of inter-reflections, changes in background, changes in environmental conditions, etc. Moreover, in situations where R contains only a part of M, or when R contains pixels not belonging to M, the intersection value will be less than 1.0. At the same time very low values of $S(R,M)$ may be caused due to partial similarity between models and/or background pixels and other noise. They do not indicate the presence of the model object. We eliminate all matches with very low values by applying a low threshold θ. It is clear that this simple thresholding alone is not sufficient, since all models with histogram intersection values above the threshold need not be present in the scene. Several models may have intersection values above the threshold θ for the same or overlapping focus regions. It has to be ensured that the regions associated with different objects are disjoint. We adopt a winner takes all policy combined with the removal of detected objects to resolve such conflicts.

A higher histogram intersection value denotes a better match between the region and the model. Let the model-focus region pair (R',M') have the highest intersection value among all the model region pairs, i.e.

$$S(R', M') = \max_{M \in M_c, R \in R_c} S(R, M).$$

Then M' has the maximum evidence for being present in R' and M' is accepted as the winner. The focus region having the highest histogram intersection value with a model is determined using active search. Active search employs upper bounds on the histogram intersection value for pruning the search area.[14] Consequently, the best matching focus region is determined by evaluating the histogram intersection of a small fraction of the focus regions. The salient aspects of active search are briefly discussed in Section 2.3.

Once a model M' and focus region R' are identified as the winning pair, we have to prevent other models from matching against the same pixels as M'. However, the exact pixels in R' which contributed to the match between model M' and R' are not known. But a large intersection value indicates that most of R' contributed to the match and the winner has a comparatively large intersection value. Therefore, we associate all the pixels of R' to the model M', and the pixels belonging to R' are masked to prevent them from being matched against other models. It may be recalled that any masked pixels are not considered while evaluating the histogram intersection. Consequently, the pixels belonging to R' do not take part in further matches.

The effect of masking pixels belonging to a focus region is schematically shown in Fig. 2. The region R_1 has no pixels in common with the masked region and hence remains unchanged. On the other hand, regions R_2 and R_3 overlap the masked region and do not constitute the entire square window. Region R_3 forms a small

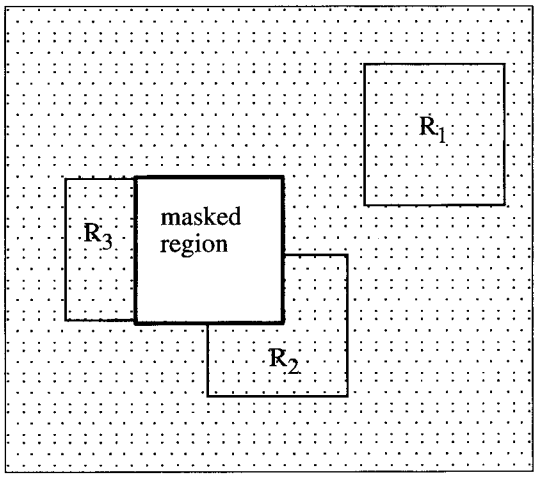

Fig. 2. The effect of masking pixels of a focus region on other focus regions in the image.

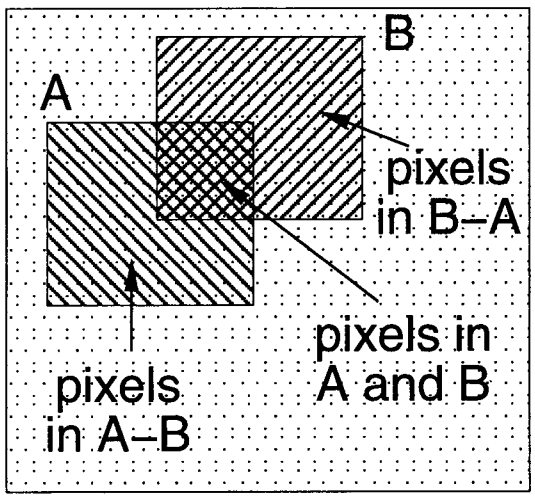

Fig. 3. Two intersecting focus regions A and B in an image.

narrow region of the input scene and its color distribution will not, in general, constitute a good feature. This effect is not restricted to a given image size but will prevail across all resized images. Also, several focus regions belonging to other resized images may also get modified as a result of masking a region. Some of them may end up having only a few unmasked pixels. Such regions also do not provide a good feature for matching. Hence all focus regions with a fraction of unmasked pixels less than some constant $\beta < 1$ are not considered in later match and prune steps. The pruned set of competing regions R'_c becomes Fig. 3:

$$\mathsf{R}'_c = \mathsf{R}_c - \{R^k_{ij} \text{ such that fraction of unmasked pixels in } R^k_{ij} > \beta\}. \tag{2}$$

It may be noted that since at least the region R' is removed from the set of competing focus regions, the set of competing focus regions strictly decreases after every match and prune step.

The set of models competing for a match are pruned based on the following observations. Consider a model M_n which is not the current winner. The histogram intersection of this model with any focus region can increase only due to masking. From equation (2) it follows that the maximum fraction of pixels in a competing focus region which may be masked is $(1-\beta)$. Consider a focus region R^{ij}_k having histogram intersection $S(R^{k_{ij}}, M_n)$. The maximum increase in $S(R^{k_{ij}}, M_n)$ due to masking will be when the masked pixels do not contribute to the histogram intersection. That is, when the total number of pixels which contribute to the histogram intersection remain the same as a result of masking. Using the upper bound derived in Section 2.3 we obtain that the maximum histogram intersection value of model M_n in later steps will be bounded by

$$\frac{1}{\beta} \max_{R^k_{ij} \in \mathsf{R}_c} S(R^k_{ij}, M_n).$$

Any model for which the above value is less than the threshold θ will not become the winner in a later step. Therefore, the set of competing models are pruned as follows:

$$\mathsf{M}'_c = \mathsf{M}_c - \{M_l \in \mathsf{M}_c \text{ and } \max_{R^k_{ij} \in \mathsf{R}_c} S(R^k_{ij}, M_l) < \beta\theta\}. \tag{3}$$

In each match and prune step one region is associated with a model and the set of focused regions as well as the set of competing models are pruned. If the pruned set of focus regions R'_c and the pruned set of competing models M'_c are not empty then the match and prune process continues with the regions in R'_c and the models in M'_c. By this process, eventually the set of competing focus regions and/or the set of competing models will become empty. Then, the iterative process of matching and pruning terminates with a set of regions associated with those models which had emerged as winners in some match and prune step.

2.3. *Active search*

In this section we give a brief discussion of active search. For the sake of brevity we present the discussions considering a single model. It is clear that neighboring focus regions in an image will have similar color histograms. Active search exploits this fact for concentrating its efforts only on focus regions having high histogram intersection with the model. The search space is pruned using upper bounds on the histogram intersection measure. By this the computational effort is greatly reduced while still retaining the optimality. The upper bound for histogram intersection is derived as follows.

Result. For any two focus regions A and B such that $|A| \geq |B|$ and any model M,

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

where $|A|$, $|B|$, $|A \cap B|$ and $|B - A|$, respectively, denote the number of pixels in A, pixels in B, pixels common to A and B and pixels in B but not in A.

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

$$S(B,M) = \sum_i \min(h^M_i, h^B_i) = \frac{\sum_i \min(|B|h^M_i, H^B_i)}{|B|}.$$

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

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

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

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

Based on the above result we can compute an upper bound $\hat{S}(B,M)$ of $S(B,M)$ as

$$\hat{S}(B,M) = \frac{\min(|A \cap B|, S(A,M)|A|) + |B - A|}{|B|}.$$

In general the focus regions A and B may belong to different image sizes. Then we use the projection of the focus regions on the original image for estimating the upper bound. Let A' and B' denote the projection of A and B, respectively, on the original image. Ignoring the sampling effects we obtain

$$S(B,M) \leq \frac{\min(|A' \cap B'|, S(A,M)|A'|) + |B' - A'|}{|B'|}. \quad (4)$$

After the histogram intersection of a focus region against the model is evaluated, the upper bounds on the histogram intersection of neighboring focus regions are estimated using equation (4). Since a given focus region falls in the neighborhood of many other regions, several upper bound estimates will be obtained for a focus region. The histogram intersection of a focus region is actually evaluated only if the least among these upper bound estimates is higher than the threshold θ and the current best match. The active search algorithm for determining the focus region having the highest histogram intersection with a model M is given below.

Algorithm Active Search

1. Set $\theta'=\theta$, and $lub(R_{ij}^k, M) = 1.0$ for all $R_{ij}^k \in R$.
2. Get the next focus region R_{ij}^k. If $\text{lub}(R_{ij}^k, M) < \theta'$ then set $S(R_{ij}^k, M) = 0$ and go to step 5.
3. Compute $S(R_{ij}^k, M)$ using algorithm Evaluate. Set $\theta' = \max\left(S(R_{ij}^k, M), \theta'\right)$.
4. Compute $\hat{S}(R_{uv}^p, M)$ for R_{uv}^p in the neighborhood of R_{ij}^k using equation (4). Set $\text{lub}(R_{uv}^p, M) = \min\left(\text{lub}(R_{uv}^p, M), \hat{S}(R_{uv}^p, M)\right)$.
5. If more focus regions are remaining go to step 2.
6. R_{uv}^p such that $S(R_{uv}^p, M) = \max_{R_{ij}^k \in R} S(R_{ij}^k, M)$ is the focus region with highest histogram intersection value, if $S(R_{uv}^p, M) > \theta$. If $S(R_{uv}^p, M) < \theta$ no focus region has histogram intersection with M higher than the threshold θ.

Figure 4 shows the center points of the focus regions for which histogram intersection against the model was evaluated by algorithm active search. In the example shown, active search matched only 361 out of more than 6093 focus regions obtained with s=4, w=32 and 10 different image sizes. It may be observed that this is much less than all 32×32 focus regions belonging to a 128×128 image. From Fig. 4, it may be observed that the search concentrates on the region where the object is actually present. Consequently, the absence of the object in an image is quickly determined after matching very few focus regions. This would be advantageous in large database retrieval tasks where several uninteresting images can be skipped quickly.

In algorithm active search, when there are several models, the upper bounds for each model M_n have to be maintained separately. Also, after one focus region is associated with a model, some pixels in the competing focus regions may be masked. In such cases the number of pixels which are masked in the respective projected regions are to be subtracted from $|A'|$, $|B'|$, $|A' \cap B'|$ and $|B' - A'|$ before applying equation (4).

The focused color intersection with active search method may be specified as follows:

Algorithm Focused Color Intersection

1. Set $\mathsf{M}_c = \mathsf{M}$ and $\mathsf{R}_c = R$ where R is defined by equation (1) and M is the set of models.
2. For each model $M \in \mathsf{M}_c$ determine the best matching focus region R^M using algorithm Active Search.
3. Let $S(R^M, M) = \max_{M \in \mathsf{M}_c} S(R^M, M)$. Associate region R^M with model M.
4. Mask all pixels belonging to focus region R^{M}. Modify all focus regions accordingly.
5. Evaluate the pruned set of focus regions R'_c and the pruned set of models M'_c following equations (2) and (3), respectively.
6. If M'_c or R'_c is empty then terminate.
7. Set $\mathsf{M}_c = \mathsf{M}'_c$, $\mathsf{R}_c = \mathsf{R}'_c$ and go to step 2.

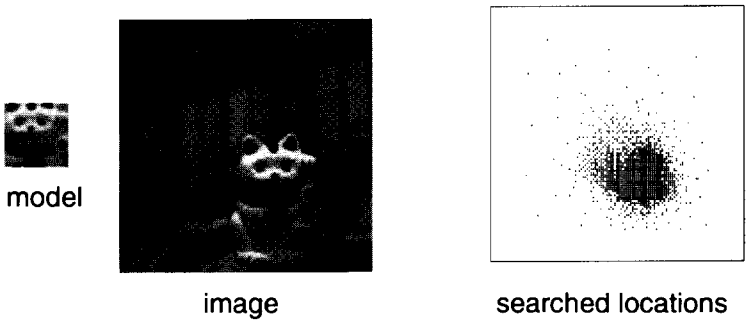

Fig. 4. The center points of focus regions matched by active search for determining the best matching focus region.

In Section 3 we present the experimental results obtained using the above algorithm.

3. EXPERIMENTAL RESULTS

Experiments were conducted with a large number of scenes under various conditions. In this section we present the results obtained for 30 scenes using 14 models. No special effort was made to segment the models from the background. However, the smallest rectangular clipping window which contains the model object was used to clip the model images. This reduces the number of background pixels present in the model image. The set of models used in the experiments is shown in Fig. 5. Each model was represented by a normalized intensity (I), hue (H) and saturation (S) histogram constructed from its RGB image.

The scenes used in the experiments consisted of one to six model objects and other objects. These images were taken in the natural environment of the laboratory. In some cases multicolored backgrounds were deliberately introduced. Three images consisted of laboratory scenes not containing any of the model objects. The objects in the scenes were kept at arbitrary orientations and were occluded by other objects. Each image was scaled to 128×128 pixels.

Histograms over the intensity (I), hue (H) and saturation (S) space were used for matching between models and focus regions. Other spaces such as RGB or LUV may also be employed for this purpose. Discussions on the characteristics of various color spaces may be found in the literature [see for example reference (17)]. The IHS space was quantized by coarse divisions along the I-axis. The S-axis was divided more finely than the I-axis and the H-axis had more divisions than S-axis. This type of division was chosen since the most important cue for color matching is hue and intensity is the least significant among the three. Moreover, the small variations in lighting conditions and reflected light from other objects in the scene could lead to changes in the intensity. This suggests that the intensity axis be coarsely divided.

The parameters α, β and θ were fixed at the following values:

$$\alpha = 0.8, \quad \beta = 0.4, \quad \theta = 0.3.$$

Changing the value of α in the range of 0.8–0.99 had no effect on the recognition rate. However, higher α implies a greater number of resized images with small differences in scale and consequently a greater number of focus regions with small differences. This would result in better quality regions extracted for an object. However, higher α would also imply more computational effort. On the other hand, lower α would reduce computation at the cost of the quality of the regions extracted.

The value of β denotes the minimum fraction of unmasked pixels in a focus region, and thereby determines the focus region pruning rate. It was observed that β in the range of 0.1–0.75 did not affect the recognition rate. However, higher β prunes more focus regions and therefore a lower number of regions may be extracted for an object than a lower β. As a result, large parts of the objects may be missed out. On the other hand, lower values of β result in less pruning of the set of focus regions and consequently more computational effort. Thus, the choice of the parameters α and β depends on the relative importance of computational efficiency and quality of regions extracted. A lower value of α and a higher value of β may be used for increased computational efficiency. A higher α and lower β may be employed for better quality of regions extracted. Moreover, with overlapping objects, it may be necessary to consider focus regions with a lower number of unmasked pixels. In such situations also a lower value of β may be used. $\beta = 0.4$ was found to be a satisfactory value.

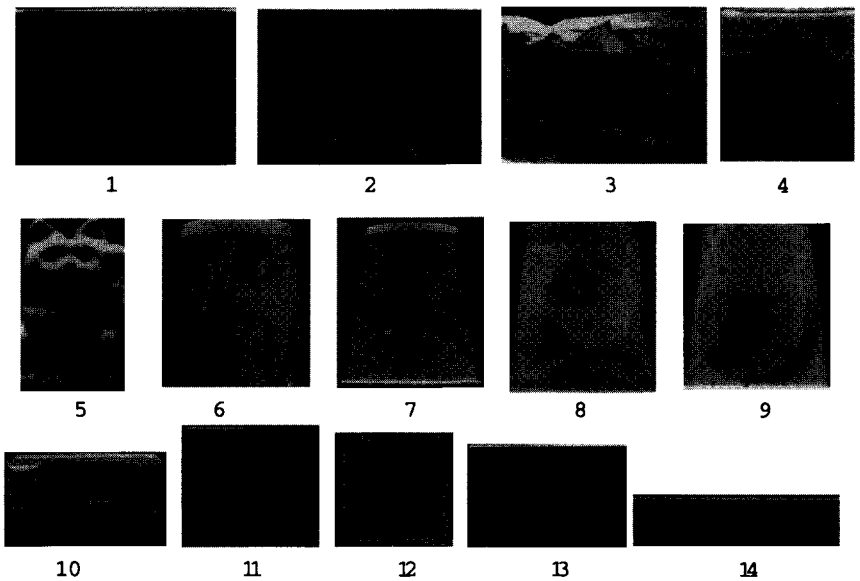

Fig. 5. The set of 14 models used in the experiments.

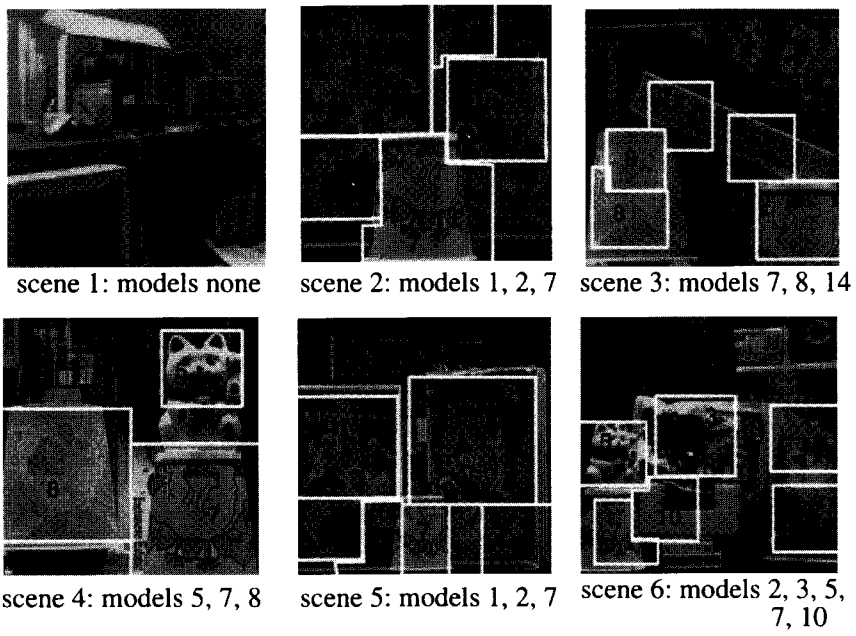

Fig. 6. Sample results of object extraction. The areas detected for each object are marked along with the corresponding model numbers.

The value of θ thresholds the histogram intersection value and determines the pruning rate for the set of competing models. It was observed that small variations in this threshold did not affect the results. However, large increases lead to more misses and large decreases lead to more false alarms. It may be possible to eliminate false alarms by a verification process. However, recovering missed objects may be difficult. Hence, lower values of θ are preferable. The value of 0.3 was experimentally found to be satisfactory.

Figure 6 shows some sample results obtained. The model and focus region histograms had 5, 50 and 40 divisions along the I, H and S axes, respectively. A window size of w=32 and shift s=4 was used for extracting the focus regions. In Fig. 6, scene 1 contains no models and none were detected. It may be pointed out that, for scenes with no models, upper bound pruning matched only less than 100 focus regions per image. In scenes 2–6, shown in Fig. 6, the regions extracted are indicated along with the corresponding model numbers. From the results, it may be observed that the method is stable against changes in the background, orientation, shape and size. The distribution of histogram intersection values of models 3, 4, 5 and 6 with scene number 6 is shown in Fig. 7. The intersection values for each focus region in the 128×128 pixel image of the scene are shown in the figure. The distribution for models 3 and 5 have a distinct peak denoting the presence of these models in the input scene. On the other hand, the distribution for models 4 and 6 do not have any clear peaks since these models are not present in the image. This shows that the color histogram intersection between models and focus regions provides an efficient discriminant for detecting and locating objects. In Fig. 8, the three-dimensional I–H–S histograms of model number 5 [marked (a)], scene number 6 [marked (b)] and the focus region of scene number 6 containing model number 5 [marked (c)] are shown. The size of the black boxes are proportional to the histogram values. From this figure it may be observed that (a) and (b) are vastly different. For example, the most prominent peaks in (a), denoted in the figure by the two larger boxes, are absent in (b). On the other hand, the histogram (c) of the focus region containing the object is quite similar to (a). That is, the focus region's histogram is quite similar to the model histogram, whereas the entire scene's histogram vastly differs from that of a model present in the scene. Consequently, methods employing whole image histogram matching fare poorly compared with focused color intersection.[13–15] Thus, with multiple objects in the scene, histogram intersection *per se* is not sufficient; focusing on parts of the scene is a must.

The effect of varying the histogram bin size, scanning window size and the number of pixels by which the window is shifted for scanning the input image were studied. The effect of changing the histogram bin size on misses and false alarms is tabulated in Table 1. The consolidated number of times a model present in the scene was not detected (misses) and a model not present was detected (false alarms) are tabulated. It is seen that very coarse divisions lead to false alarms. This arises because under coarse division of the color space different colors get mapped to the same bin leading to false matches. Similarly it is seen that finer divisions, particularly along the I-axis, lead to more miss errors. This is expected since with finer divisions even small variations due to lighting or other interferences will map a pixel to a different histogram bin. Since the intensity component is

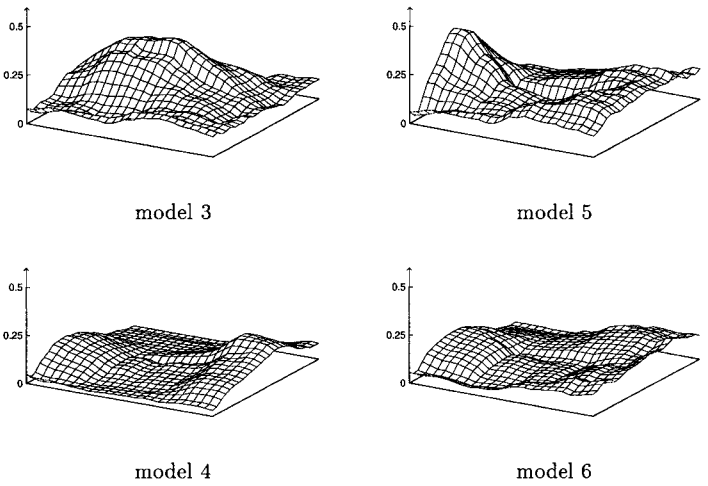

Fig. 7. The histogram intersection values of models 3, 4, 5 and 6 with focus regions obtained from scene 6 at 128×128 pixels.

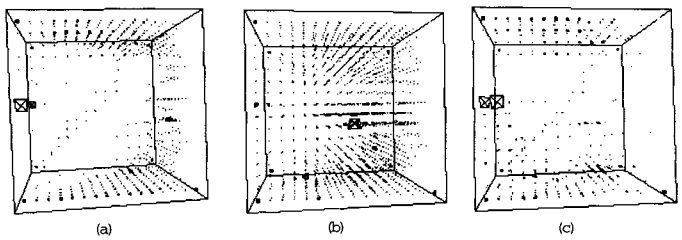

Fig. 8. The I–H–S histogram for (a) the model numbered 5, (b) the input scene numbered 4 and (c) the focus region of input scene 4 containing an instance of model number 5.

Table 1. Effect of changing the number of histogram bins

Divisions along			Misses	False alarms
I	H	S		
5	30	20	0	7
5	40	30	0	1
5	40	30	0	0
8	40	30	0	0
10	50	40	8	0

Table 2. Effect of changing the size of the scanning window

Divisions along			Window size	Misses	False alarms
I	H	S			
5	40	30	24×24	0	5
5	40	30	32×32	0	1
5	40	30	40×40	0	1
5	50	40	24×24	3	2
5	50	40	32×32	0	0
5	50	40	40×40	0	0

most influenced by lighting variations and interference from other objects in the scene, coarse divisions along the I-axis are recommended. However, from the results we observe that small changes in the bin size (as seen from rows 2, 3 and 4 of Table 1) do not affect the results very much.

In Table 2 we present the effect of varying the scanning window size. From the table it is observed that increasing the window size from 32×32 to 40×40 does not change the results. However, a reduction in the window size to 24×24 leads to increased false alarms and misses. This occurs since a smaller window size would be focusing on a small portion of the object. The color distribution of a small portion of the object may not provide enough information for distinguishing it. When the image is sufficiently reduced in size for a small window to cover a larger part of the image, the effects of sampling would affect the results. Thus, larger window sizes would be suitable. This is advantageous since larger windows will reduce the number of focus regions.

In Table 3 we present the effect of changing the number of pixels by which the window is shifted while scanning the image. A change from 4 to 16 pixels has a negligible influence on the results. Only in one instance was an object missed when the shift was increased to 16 pixels. Since color distributions do not vary much for small changes in the focus region, larger shifts do not lose

Table 3. Effect of shifting the window by 4, 8 and 16 pixels for input image scanning

Divisions along			Window shift	Misses	False alarms
I	H	S			
5	40	30	4	0	1
5	40	30	8	0	1
5	40	30	16	0	1
5	50	40	4	0	0
5	50	40	8	0	0
5	50	40	16	0	0

information for detecting the objects. However, the region extracted for a model present in the image will vary with changes in the number of pixels by which the scanning window is shifted. The extracted regions will be better with smaller shifts than with larger shifts.

Thus, if the objective is only to identify the presence of the object then large window sizes along with large shifts may be employed. On the other hand, if the extracted regions are of importance then a smaller window size with smaller values of shifts and a larger value of α close to 1.0 combined with a lower value of β need to be employed. This would, however, be computationally more expensive than employing larger window sizes and shifts, lower α and higher β.

4. CONCLUSION

In this paper, we have proposed a focused color intersection method with efficient searching for identifying and extracting known objects from a complex scene using only color distributions. An efficient algorithm for evaluating color histogram intersection between a model and a focus region in the image has been presented. This algorithm's complexity is independent of histogram size. Hence it can be employed for fast matching with high-dimensional histograms such as co-occurrence histograms and those obtained from multiband data.

An upper bound pruning strategy called active search has been proposed for efficiently searching the focus regions. Active search results in huge reduction in computational effort,[14] without sacrificing accuracy. The search concentrates only on focus regions having high similarity with the object. Therefore, the absence of an object is quickly determined after matching only a few focus regions. This nature is especially useful for skipping over uninteresting images in database retrieval operations. Investigations are being conducted for adapting this search strategy to other features and combinations of features.

The experimental results demonstrate that the method correctly identifies the objects and extracts the regions occupied by the objects in the scene. The method works well under complex backgrounds, occlusion, changes in two-dimensional rotation, shape and scale. It is fairly stable for small variations in three-dimensional orientation. However, large changes in three-dimensional orientation could lead to changes in the color distribution and adversely affect the results. Matching against multiple three-dimensional views of the models could overcome this. More robust color matching techniques are being investigated to accommodate illumination changes.

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

REFERENCES

1. W. E. L. Grimson, *Object Recognition by Computer: The Role of Geometric Constraints*. MIT Press, Cambridge, Massachusetts (1990).
2. P. Suetens, P. Fua and A. J. Hanson, Some computational strategies for object recognition, *Surveys* **24**(1), 5–62 (March 1992).
3. A. Rosenfeld and A. C. Kak, *Digital Picture Processing*. Academic Press, New York (1976).
4. H. Murase and S. K. Nayar, Image spotting of 3d objects using parametric eigenspace representation, in *Proc. Ninth Scandinavian Conf. on Image Analysis* (June 1995).
5. M. J. Swain and D. H. Ballard, Color indexing, *Int. J. Comput. Vision* **7**(1), 11–32 (November 1991).
6. M. Stricker and M. Swain, The capacity of color histogram indexing, in *CVPR94*, pp. 704–708 (1994).
7. J. Hafner, H. S. Sawhney, W. Equitz, M. Flickner and W. Niblack, Efficient color histogram indexing for quadratic form distance functions, *IEEE Trans. Pattern Analysis Mach. Intell.* **17**(7), 729–736 (1995).
8. M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Yonkani, J. Hafner, D. Lee, D. Petkovic, D. Stede and P. Yanker, Query by image and video content: The QBIC system, *IEEE Comput.* **28**(9), 23–32 (September 1995).
9. B. M. Mehtre, M. S. Kankanhalli, A. D. Narasimhalu and G. C. Man, Color matching for image retrieval, *Pattern Recognition Lett.* **16**, 325–331 (1995).
10. J. K. Wu, A. D. Narasimhalu, B. M. Mehtre, C. P. Lam and Y. J. Gao, CORE: A content-based retrieval engine for multimedia information systems, *ACM Multimedia Systems* **3**(1), 25–41 (1995).
11. E. Binaghi, I. Gagliardi and R. Schettini, Image retrieval using fuzzy evaluation of color similarity, *Int. J. Pattern Recognition Artif. Intell.* **8**(7), 945–967 (August 1994).
12. R. Schettini, Multicolored object recognition and location, *Pattern Recognition Lett.* **15**, 1089–1097 (1994).
13. C. Hashizume, V. V. Vinod and H. Murase, Focussed color matching for object retrieval from movies, in *Proc. 1996 IEICE Spring Conf.*, pp. D–352 (January 1996).
14. V. V. Vinod, H. Murase and C. Hashizume, Focussed color intersection with efficient searching for object detection and image retrieval, in *Proc. IEEE Conf. on Multimedia Computing Systems*, pp. 229–233 (June 1996).
15. V. V. Vinod and H. Murase, Object location using complementary color features: Histogram and DCT, in *Proc. ICPR'96*, pp. A:554–559 (August 1996).
16. F. Ennesser and G. Medioni, Finding waldo, or focus of attention using local color information, *IEEE Trans. Pattern Analysis Mach. Intell.* **17**, 805–809 (1995).
17. M. Stokes, M. D. Fairchild and R. S. Berns, Precise requirements for digital color reproduction, *ACM Trans. Graphics* **11**, 406–422 (1992).

About the Author — V. V. VINOD received the B.Tech. (Hons.), M.Tech and Ph.D. degrees in Computer Science and Engineering from Institute of Technology, Kharagpur, India, in 1988, 1990 and 1994, respectively. He has worked at the Indian Institute of Technology, Kharagpur (1989–1993), Electronics Research and Development Center, Trivandrum (1993–1994) and Ashok Leyland Information Technology Ltd., Bangalore (1994–1995). During 1995–1997 he was with NTT Basic Research Laboratories, Nippon Telegraph and Telephone Corporation, Japan, where he worked on image retrieval and video analysis. He is currently with the Institute of Systems Science, National University of Singapore. His research interests include neural networks, genetic algorithms, image processing, pattern recognition and video analysis.

About the Author — 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, 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, and IPSJ.