Focussed Color Intersection for Object Extraction from Cluttered Scenes

V V Vinod Hiroshi Murase

NTT Basic Research Labs, 3-1 Morinosato Wakamiya
Atsugi-shi, Kanagawa, 243-01 Japan
e-mail {vinod,murase}@apollo3.brl.ntt.jp

Abstract

In this paper, we present focussed color intersection, a method for extracting colored objects from a cluttered scene. The method efficiently matches parts of the scene and competitively associates them to the models. Experiments conducted demonstrate that multiple known objects in complex scenes can be extracted by this process. The method is stable against scale changes, 2-dimensional rotation, moderate changes in shape and partial occlusion.

1 Introduction

Extracting instances of models from cluttered scenes has several practical applications such as constructing model based index for image databases, visual querying, target selection etc. Most of the approaches to object recognition and location make use of geometric features [3, 9]. Simple geometric features requires costly matching procedures and complex geometric features are difficult to extract. Moreover, complex and computationally expensive procedures are often required to take into account variations in view point, shape etc. Template matching using features other than geometric features have been successfully attempted as an alternative [6, 7, 9]. Usually several views of the model are stored and matched against the scene to account for changes due to 2 or 3 dimensional orientation, scaling, shape changes etc. It would be desirable to do the matching using features invariant to as many changes as possible. The color distribution of an object is stable to changes in 2-dimensional orientation, moderate changes in shape, partial occlusion, and small changes in object pose.

The role of color, as an efficient cue for matching images, has been demonstrated in several recent works [1, 4, 5, 8, 10]. Color histogram matching is commonly employed for image indexing and content based image retrieval applications [2, 12]. Usually, the similarity of the image histogram and model histogram is exploited for indexing into image databases and retrieving similar images. However, the image and model histograms will be vastly different when the model occupies only a small portion of the image. In such situations, matching the image and model histograms will not provide any useful information for object detection. At the same time, the model histogram will be similar to the histogram of that part of the image containing the model. An appropriate color histogram matching strategy which focuses its search on different parts of the image would work well in such situations.

In this paper, we propose focussed color intersection, an iterative method which matches the model against parts of the scene. The similarity measure employed is histogram intersection proposed by Swain and Ballard [10]. The histogram intersection between model and focus regions are efficiently evaluated without explicitly computing the histograms of the focus regions. A competitive matching process identifies the models present and prunes the set of models and focus regions. The proposed method can detect colored objects against complex colorful backgrounds, irrespective of size, 2-dimensional orientation and partial occlusion. It is, however, influenced by changes in lighting conditions and object pose. The focussed color intersection method is described in section 2 followed by the algorithm in section 3. The experimental results are given in section 4 and the conclusions in section 5.

2 Focussed Color Intersection

The following setting is considered for developing the method.

> Given a set of models $\mathsf{M} = \{M_n\}$, $n = 1, \ldots, N$ where each M_n is the color image of a known object and an $X \times Y$ pixel color image $\mathcal{I}$ of a scene ($\mathcal{I} = \mathbf{p}_{xy}, x = 1, \ldots, X, y = 1, \ldots, Y$), identify any instances of the models present in the image and extract the regions occupied by them.

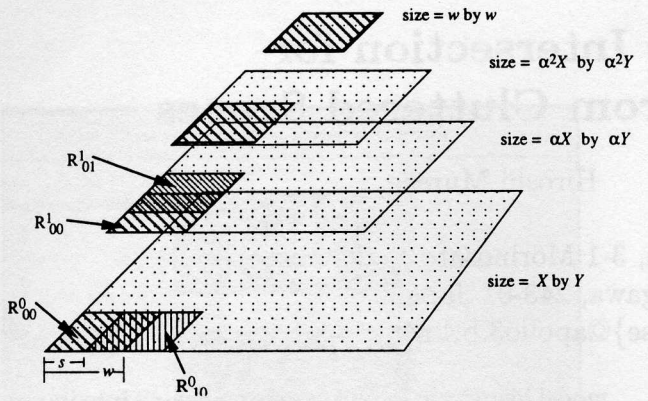

Figure 1: The focus region extraction process.

We shall refer to the image $\mathcal{I}$ as scene or image. The image $\mathcal{I}$ 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 2 dimensional orientation and small changes in object pose. Objects may be partially occluded and the shape of an (non-rigid) object may vary from scene to scene.

The method consists of three steps. The first step constructs a set of focus regions of the given image. The second step is matching and pruning where the focus regions are efficiently matched against the models and assigned to a model one by one. After every assignment of a focus region to a model, the set of focus regions and the set of competing models are pruned. This process of matching and pruning continues till no more focus regions can be assigned to any model. In the third step a confidence measure for each model is computed taking into account all the regions assigned to it.

2.1 Extracting Focus Regions

The histogram of the entire scene will be quite different from that of any of the models when the scene consists of several objects. Hence, the matching process has to focus on parts of the scene. Ideally, the focus region should be such that it contains a single object. In general, it would not be possible to ensure this and we will have to consider all sizes and locations of the objects. Since the size of the object may vary from scene to scene the focus regions should cover small parts of the scene as well as the whole scene. Now, the color distribution of very small regions in the image will not carry any effective information. Therefore, the focus regions should cover a sufficiently large part of the image.

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. We consider a square shape and focus regions are extracted using a square window of size $w \times w$ pixel. Different focus regions are extracted by scanning the image with this window. For scanning the image, the window is shifted by s pixel in one direction at a time. After one complete scan is over, the 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 extracting larger regions from the original image for the identification process to focus upon. This will account for variations in object size. This process of resizing by a factor of α and scanning the resized images is continued until the image becomes smaller than the scanning window. The focus region extraction process is depicted in figure 1. The original image and three resized images are shown in the figure. 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 complete set of focus regions extracted by the above process may be characterized as follows. Let $\mathcal{I}^k$ denote the image resized by α^k and $\mathbf{p}^k_{xy}$ denote the pixel belonging to $\mathcal{I}^k$. Then

$$\mathcal{I}^k = \mathbf{p}^k_{xy} \quad x = 1, \ldots, \alpha^k X \quad y = 1, \ldots, \alpha^k Y$$
$$\text{where } \mathbf{p}^k_{xy} = \mathbf{p}_{uv}, \quad u = \left\lfloor \frac{x}{\alpha^k} \right\rfloor \quad v = \left\lfloor \frac{y}{\alpha^k} \right\rfloor$$

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

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

Any of the regions in R could contain a part or whole of any model in M. Let M_c and R_c denote the set of competing models and focus regions respectively. Then $\mathsf{M}_c = \mathsf{M}$ and $\mathsf{R}_c = \mathsf{R}$.

2.2 Matching and Pruning

In the matching and pruning steps we identify maximally similar focus region-model pairs such that the focus regions are disjoint. The similarity of the match between a focus region and a model is evaluated using the normalized histogram intersection

measure proposed by Swain and Ballard [10]. When we refer to histogram intersection, it shall denote normalized histogram intersection. The normalized histogram intersection of a model M_n and a focus region R_{ij}^k shall be denoted by $I(M_n, R_{ij}^k)$.

In the first match and prune cycle the histogram intersection of each focus region in R_c, against each model in M_c, is computed. In the case of a perfect match between M_n and R_{ij}^k, $I(M_n, R_{ij}^k)$ will be equal to 1.0. However, in general, even if R_{ij}^k contains exactly the same object as M_n, the histogram intersection would be less than 1.0 due to reflected light, environmental noise etc. In situations where R_{ij}^k contains only a part of M_n, or when R_{ij}^k contains pixel not belonging to M_n etc., the intersection value will be quite less than 1.0. At the same time very low values of $I(M_n, R_{ij}^k)$ may be caused due to partial similarity between models and/or background pixel and other noise. These do not indicate the presence of the model object. All model-focus region pairs with very low values of histogram intersection are eliminated by applying a threshold θ. However, all histogram intersections above θ do not necessarily indicate the presence of the model in the scene. It may be that several models have intersection value above θ for the same or overlapping focus regions. That is, several models are competing for a match with the same part of the scene. Then only one model should be recognized. However, if two models have high intersections values against disjoint focus regions then both should be identified. We achieve this by identifying only one model-focus region pair at a time and then preventing all pixel in that focus region from matching against other models. Since higher histogram intersection value indicates a better match, the focus region-model pair with the highest value of histogram intersection is accepted as the winner. That is, R_{uv}^p is assigned to model M_l if $I(M_l, R_{uv}^p) = \max_{M_n \in \mathsf{M}_c, R_{ij}^k \in \mathsf{R}_c} I(M_n, R_{ij}^k)$.

Pixels which match M_l should be prevented from being matched against other models. However, the exact pixels in R_{uv}^p which contributed for the match between model M_l and R_{uv}^p is not known. But, a large intersection value indicates that most of R_{uv}^p contributed to the match, and, the winner has a comparatively large intersection value. Therefore we associate all the pixel in the area covered by of R_{uv}^p to the model M_l. These pixels are masked and not considered in future histogram intersection evaluations. Consequently, the pixels belonging to R_{uv}^p do not take part in further matches.

As a result of masking R_{uv}^p, the intersection values of the models against several of the remaining focus regions may change. The effect of masking is schematically shown in figure 2. The masked region and three other focus regions R_1, R_2 and R_3 are shown in the figure. The region R_1 has no pixel in common with the masked region. Hence, the histogram intersection values of this region against the models remain unchanged and need not be computed again. On the other hand regions R_2 and R_3 overlap the masked region. For such regions the histogram intersection values have to be recomputed. Now, R_3 contains very few unmasked pixel and its color distribution will not, in general, constitute a good feature and need not be considered for matching against the models. The effect of masking prevails across all image sizes and is depicted in figure 3. In figure 3, the masked region belongs to image of size $\alpha X \times \alpha Y$. This is equivalent to masking a larger areas from larger images and smaller areas from smaller images. Several focus regions belonging to other resized images also get modified as a result of masking a region. Some of them may end up having only few unmasked pixel. Such regions do not provide a good feature for matching. Hence all regions for which the fraction of unmasked pixel is less than some constant $\beta < 1$ are removed from the set of competing regions. Thus the set of focus

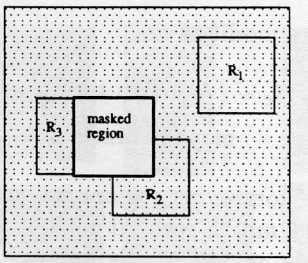

Figure 2: The effect of masking on the same image

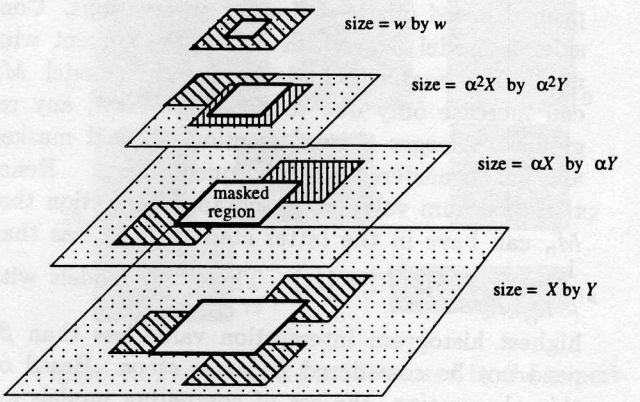

Figure 3: The effect of masking, across images

regions are modified as

$$\mathsf{R}'_c = \mathsf{R}_c - \{R^k_{ij} \text{ such that fraction of masked pixels in } R^k_{ij} > \beta\} \quad (2)$$

The set of competing regions are pruned in this way after every match step. It may be noted that since at least the region R^p_{uv} is removed from the set of competing focus regions, this set strictly decreases after every match and prune step.

The set of models competing for a match can be pruned based on the following observations. Consider a model M_n which is not the current winner. The match confidence for this model M_n can increase only due to masking. Now, any region with larger than β fraction of pixel masked are not considered in subsequent steps. Hence the maximum value of histogram intersection that M_n can have in the latter steps will be less than $\frac{1}{\beta} \max_{R^k_{ij} \in \mathsf{R}_c} I(M_n, R^k_{ij})$ [11]. Therefore, models with highest histogram intersection value less than $\beta\theta$ need not be considered in latter steps. Based on this observation, the set of competing models are pruned as follows.

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

Thus, in each match and prune step one region is assigned to a model and the set of focussed regions and the set of competing models are pruned. After pruning, the histogram intersection needs to be evaluated only for those focus regions in R'_c which have changed due to masking and models in M'_c. The process of matching and pruning continues as long as M'_c and R'_c are not empty and some model-focus region pair has histogram intersection above θ. This iterative process will eventually terminate with focus regions assigned to those models which had emerged as winners in some match and prune step.

2.3 Confidence Evaluation

Focus region identification followed by matching and pruning will yield a set of regions assigned with models detected in the scene as well as a match confidence for each region. However, the individual match confidences of the regions assigned to a model are only indicative, and not an absolute measure for the presence of the model. For example, the match value between the union of two disjoint regions belonging to a partially occluded object and the model may be higher than that of either of the regions considered individually. Similarly individual regions may have high confidence values but put together they may not resemble any model. Thus, the confidence measure will have to consider the totality of all regions assigned to a model. Once the match and prune cycle is over the histogram intersection between the model and the union of all focus regions assigned to that model is computed. This histogram intersection value serves as absolute match confidence for a particular model.

It may be mentioned that other more elaborate techniques could also be applied for evaluating the final confidence measure. This becomes considerably easier since the region extracted consists mostly of pixel belonging to one object only thereby giving indications about location, scale and orientation of the object.

3 Algorithm

In this section we present the focussed color intersection algorithm and the algorithm for evaluating the histogram intersection between a focus region and a model without explicitly computing the focus region's histogram. Each model in the set of models is represented by its normalized color histogram. For efficient computation of the histogram intersection of focus region and model each image is converted to an internal representation before applying the algorithm. In the internal representation of the image each pixel is represented by the histogram bin to which it maps. Thus, a pixel $\mathbf{p}_{xy}$ has an integral value denoting the histogram bin to which that pixel is mapped. Any color space transformations or special quantization schemes are taken into account while this representation is computed.

The focussed color intersection algorithm may be specified as below.

1 Assign $\mathsf{M}_c = \mathsf{M}$ the set of models and $\mathsf{R}_c = \mathsf{R}$ defined by equation 1.

2 Compute $I(M_n, R^k_{ij})$ for all $R^k_{ij} \in \mathsf{R}_c$ and $M_n \in \mathsf{M}_c$.

3 Let $I(M_l, R^p_{uv}) = \max_{M_n \in \mathsf{M}_c, R^k_{ij} \in \mathsf{R}_c} I(M_n, R^k_{ij})$.
If $I(M_l, R^p_{uv}) < \theta$ go to step 8.

4 Assign region R^p_{uv} to model M_l Mask all pixel in the area covered by R^p_{uv}.

5 Evaluate R'_c and M'_c following equations 2 and 3 respectively.

6 If M'_c or R'_c is empty go to step 8.

7 Assign $M_c = M'c$ and $R_c = R'_c$. Compute $I(M_n, R^k_{ij})$ for all $M_n \in M_c$ and $R^k_{ij} \in R_c$ which have changed due to masking. Go to step 3.

8 For each model M having at least one region associated with it compute $I(M, R_1 \bigcup R_2 \bigcup \ldots R_m)$ where $R_1, \ldots, R_m$ are the regions associated with M.

In the above algorithm the computationally intensive operation is to evaluate the histogram intersections between models and focus regions. The histogram intersection of two histograms H and H', is defined as $\sum_i \min(H_i, H'_i)$ where the sum is taken over all histogram bins 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 model histogram. This would require one scan over the focus region and at least one scan over the histogram. The resulting complexity will be of order of $\max(w^2, b)$ where b is the number of bins in the histogram. Hence finer quantization of the color space will result in more computation time. However, since the focus region has only w^2 pixels, there will be utmost 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 present the following algorithm for computing the normalized histogram intersection of focus region R^k_{ij} and a model M_n. The histogram intersection is computed from the internal representation of the image without actually constructing the histogram of R^k_{ij}.

1 Focus region R^k_{ij}, Model histogram H^n, Temporary histogram H.

2 Initialize count=0, $I(M_n, R^k_{ij}) = 0$

3 For each pixel $\mathbf{p}^k_{xy}$ of R^k_{ij} do
$H_{\mathbf{p}^k_{xy}} = 0$ if $\mathbf{p}^k_{xy}$ is not masked
count = count + 1 if $\mathbf{p}^k_{xy}$ is masked

4 For each pixel $\mathbf{p}^k_{xy}$ of R^k_{ij} do
If $(H_{\mathbf{p}^k_{xy}} < H^n_{\mathbf{p}^k_{xy}})$ then

$$H_{\mathbf{p}^k_{xy}} = H_{\mathbf{p}^k_{xy}} + \frac{1}{\text{count}}$$

$$I(M_n, R^k_{ij}) = I(M_n, R^k_{ij} + \frac{1}{\text{count}}$$

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. Thus, the algorithm constitutes an efficient method to compute histogram intersections of the models against focus regions in the image.

Figure 4: The set of 14 models used in the experiments

4 Experimental Results

Experiments were conducted for a large number of images under various conditions. In this section we present the results obtained for 30 images using fourteen models. While constructing the model set no special effort was made to segment the models from the background. The smallest rectangular clipping window which contains the model object was used to clip the model's image. This reduces the number of background pixel present in the model's image. The set of models used in the experiments are shown in figure 4.

The scenes used in the experiments consisted of one to six model objects and other objects. The scene images were taken in the natural background of the laboratory. In some cases multicolored backgrounds were deliberately introduced. Three images consisted of laboratory scenes with no models. The objects in the scenes were kept at arbitrary orientations and was occluded by other objects. Each scene image was of 128x128 pixels.

Histograms over the intensity (I), hue (H) and

saturation (S) space was used for matching between models and focus regions. The IHS space was quantized by coarse divisions along the I-axis. The S-axis was divided more finely than I-axis and the H-axis had more divisions that S-axis. This was motivated by the fact, that, hue is the most important cue for color matching and intensity is least significant. Moreover, coarse quantization along I-axis will mean less susceptibility to intensity variations. Experiments were conducted with different quantizations of the IHS color space, scanning window size w and the number of pixels s by which the window was shifted for scanning the images. For each choice of histogram quantization, the normalized histogram of each model and the internal representation of each image was constructed with the chosen quantization levels.

The parameters α, β and θ were chosen as $\alpha = 0.8$, $\beta = 0.4$ and $\theta = 0.3$. The choice of factor $\alpha = 0.8$ resulted in successive image sizes of 128x128, 102x120, 81x81, 64x64, 51x51, 40x40, 32x32 and 24x24 pixel. Changing the value of α in the range of 0.8 to 0.99 had not effect on the recognition rate. Higher α implies more number of resized images with small differences in scale and consequently more number of focus regions with small differences. This would result in better quality of 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.

It was observed that β in the range of 0.1 to 0.75 did not effect the recognition rate. Higher β prunes more focus regions and therefore lesser number of regions may be extracted for an object than a lower β. Lower values of β results in less pruning of the 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 higher value of β may be used for increased computational efficiency. And a higher α and lower β may be employed for better quality of regions extracted. $\beta = 0.4$ was found to be a satisfactory value.

The value of θ thresholds the histogram intersection value and along with β 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. False alarms may be eliminated at the confidence evaluation stage, but misses cannot be recovered. Hence lower values of θ are preferable.

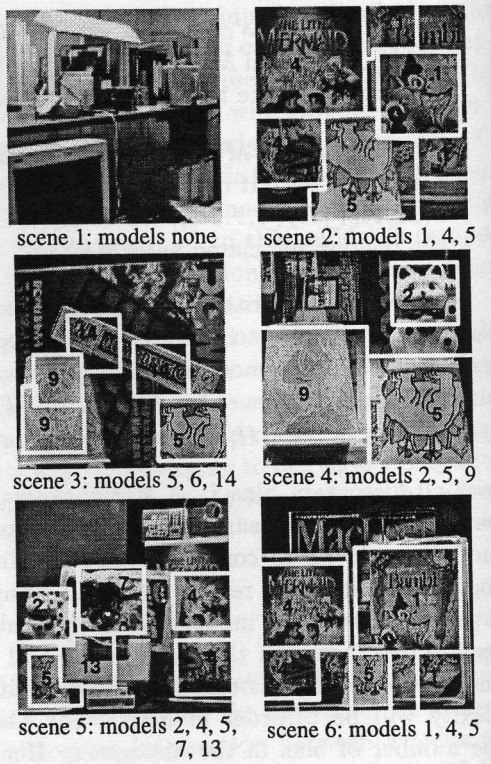

Figure 5: Results obtained for 6 scenes. The regions extracted are outlined and the corresponding model numbers are indicated

The value of 0.3 was experimentally found to be satisfactory.

Figure 5 show the results obtained for 6 scenes. The histograms were constructed with 5, 50 and 40 divisions along the I, H and S axes respectively. The other parameters were $w = 32$ and $s = 4$. That is, the focus regions were generated by scanning the images with a 32×32 window shifted by 4 pixel along one direction at a time. Scene 1 in figure 5 contain no models and none was detected. In the other cases, the extracted region boundaries and the numbers of the models assigned are shown. 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 2, 3, 7 and 8 with the 128x128 pixel image of scene numbered 7 is shown in figure 6. The distribution for models 2 and 7 have a distinct peak denoting the presence of these models in the scene. On the other hand the distribution for models 3 and 8 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.

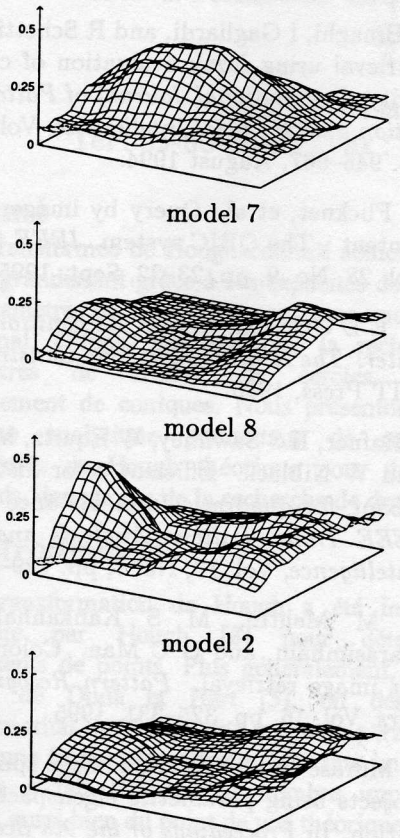

Figure 6: The histogram intersection values of models 2, 3, 7 and 8 for scene 7 at 128x128 pixels

In figure 7 the 3-dimensional IHS histograms of (a) model number 2, (b) scene number 7, and (c) the focus region of scene 7 containing model 5 are shown. In figure 7 the size of the 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 the model. Thus, when multiple objects are present in the scene, histogram intersection per se is not sufficient, focusing on parts of the scene is a must.

The consolidated number of times a model object present in the scene was not detected (misses) and a model object not present was detected (false alarms) for different histogram quantization are given in table 1. Under coarse quantization of the color space, different colors get mapped to the same bin leading

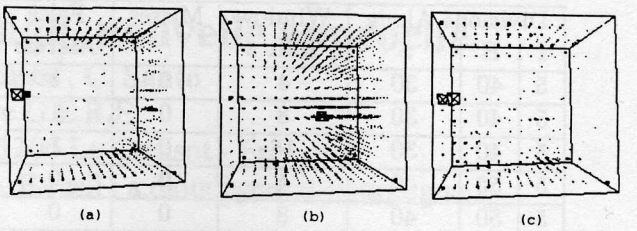

Figure 7: The IHS histogram for (a) the model number 5, (b) the scene number 4 and (c) the focus region of the scene 4 containing model 5

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

Table 1: Effect of changing histogram quantization

to increased false alarms. On the other hand, finer divisions, particularly along the I-axis, mapped similar colors to different bins leading to more misses. Since the intensity component is most influenced by lighting and interference from other objects in the scene, coarse divisions long I-axis is recommended. However, from the results we observe that small changes in the bin size do not affect the results much.

In table 2 we present the effect of varying the scanning window size. It was observed that increasing the window size from 32x32 to 40x40 does not change the results. However, a reduction in the window size to 24x24 resulted in increased false alarms and misses. This occurs since the color distribution of small parts of the object may not provide enough information for distinguishing it. And, when the image is sufficiently reduced in size for a small window to cover a larger part of the image, the effects of sampling affects the results.

In table 3 we present the effect of changing s, the number of pixel by which the window is shifted for

Divisions Along			Window	Misses	False
I	H	S	Size		Alarms
5	40	30	24x24	0	5
5	40	30	32x32	0	1
5	40	30	40x40	0	1
5	50	40	24x24	3	2
5	50	40	32x32	0	0
5	50	40	40x40	0	0

Table 2: Effect of changing scanning window size

Divisions Along			Window	Misses	False
I	H	S	Shift s		Alarms
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	1	0

Table 3: Effect of shifting the window by 4, 8 and 16 pixel for constructing focus regions

scanning the image. A change from 4 to 16 pixel had negligible effect on the results. Color distributions do not vary much for small changes in the focus region. Hence information for detecting the objects is not lost by using large s. However, the extracted regions will be better with smaller values of s.

From the results and discussions in this section we can conclude that, if the objective is only to identify objects then large w and large s with lower α and higher β may be used. If the extracted regions are of importance then a smaller w and s with α close to 1.0, and a lower value of β should be used. This latter would, however, be computationally more expensive than the former. The computational cost of using higher α and lower s can be substantially reduced by efficient searching [11].

5 Conclusion

In this paper we have proposed focussed color intersection, a method for identifying and extracting known objects from a complex scenes. Focussed color intersection efficiently associates models with disjoint regions in the image based on color similarity of the regions and the model. The experimental results demonstrate that the proposed method correctly identifies the objects and extracts the regions occupied by the objects in the scene.

Matching against multiple views of 3-dimensional objects and incorporating color constancy will improve the performance of the method. These aspects and techniques for improved confidence measure evaluation incorporating other features are under investigation.

Acknowledgments The authors wish to thank Dr. T. Ikegami, Dr. K.Ishii, Dr. N.Hagita and Dr. S.Naito of NTT Basic Research Labs for their help and encouragement in conducting this research.

References

[1] E Binaghi, I Gagliardi, and R Schettini. Image retrieval using fuzzy evaluation of color similarity. *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 8, No. 7, pp. 945–967, August 1994.

[2] M Flickner, et al. Query by image and video content : The QBIC system. *IEEE Computer*, Vol. 28, No. 9, pp. 23–32, Sept. 1995.

[3] W E L Grimson. *Object Recognition by Computer: The Role of Geometric Constraints.* The MIT Press, 1990.

[4] 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 and Machine Intelligence*, Vol. 17, No. 7, pp. 729–736, 1995.

[5] B M Mehtre, M S Kankanhalli, A D Narasimhalu, and G C Man. Color matching for image retrieval. *Pattern Recognition Letters*, Vol. 16, pp. 325–331, 1995.

[6] H Murase and S K Nayar. Image spotting of 3d objects using parametric eigenspace representation. In *Proceedings of the 9th Scandinavian Conference on Image Analysis*, June 1995.

[7] A Rosenfeld and A C Kak. *Digital Picture Processing.* Academic Press, 1976.

[8] R Schettini. Multicolored object recognition and location. *Pattern Recognition Letters*, Vol. 15, pp. 1089–1097, 1994.

[9] P. Suetens, P. Fua, and A.J. Hanson. Some computational strategies for object recognition. *Surveys*, Vol. 24, No. 1, pp. 5–62, March 1992.

[10] M J Swain and D H Ballarad. Indexing via color histograms. In *Proc. Image Understanding Workshop*, pp. 623–630, 1990.

[11] V V Vinod, H Murase, and C Hashizume. Focussed color intersection with efficient searching for object detection and image retrieval. Proc. of IEEE Conference on Multimedia, June 1996.

[12] J K Wu, A D Narasimhalu, B M Mehtre, C P Lam, and Y J Gao. CORE: a content-based retrrieval engine for multimedia information systems. *ACM Multimedia Systems*, Vol. 3, No. 1, pp. 25–41, 1995.