

A Colour Object Search Algorithm

Paul A. Walcott and Tim J. Ellis
Centre for Information Engineering
Department of Electrical, Electronic and Information Engineering,
City University, London EC1V 0HB, UK
[p.a.walcott|t.j.ellis]@city.ac.uk

Abstract

In this paper a colour object search algorithm is presented. Given an image, areas of interest are generated (for each database model) by isolating regions whose colours are similar to model colours. Since the model may exist at one or more of these region locations, each is examined individually. At each region location the object size is estimated and a growing process initiated to include all pixels with model colours. Growing is terminated when a match measure (based on object size and the number of pixels with each model colour) is maximised; if it exceeds a predefined threshold then the object is assumed to be present. Several experiments are presented which demonstrate the algorithm's robustness to scale, affine object distortion, varying illumination, image clutter and occlusion.

1 Introduction

Object search requires two capabilities: the ability to recognise an object when it comes into view and a mechanism that brings the object into view. The first of these problems (object recognition) has received a great deal of attention, but geometric solutions, e.g. the interpretation tree [1] and geometric invariance [7] have dominated. More recently, colour-based recognition algorithms e.g. colour histogram intersection [10], the colour region adjacency graph [6] and methods which use the statistics of colour space components [9] have become more popular. The second problem (bringing the object into view) has received less attention. The most straightforward solution is linear search which examines the entire search space at a high spatial resolution, however this process is time-consuming. An alternative is the use of a visual cue such as colour which is: (a) used by the human visual system; (b) salient [3]; (c) resilient to changes in spatial resolution [2] — object geometry is less reliable at low spatial resolutions.

There are several methods of performing object search using colour. Wixson and Ballard [15] used a camera mounted on a robot arm to examine the walls of a room (in real time); and for each camera gaze a confidence, based on the ratio of colour populations, was calculated to determine object presence. Swain and Ballard [10] proposed a method called histogram backprojection which determined a confidence value for each image pixel; peaks in the confidence space after smoothing corresponded to object hypotheses. Vinod et al. [11] used histogram backprojection to identify object hypotheses and colour histogram intersection [10] to verify object presence. Schettini [8] first performed a search for an object with a similar shape then used colour histogram intersection for object colour match verification. Finally, Matas et al. [6] used a colour adjacency graph (whose nodes represent model colours and

edges encode information about the adjacency of colours and their reflectance ratios) to identify object hypotheses and the colour region adjacency graph for object match verification.

All of these techniques suffer from limitations. Wixson and Ballard's colour proportion ratios are not robust to object occlusion. Histogram backprojection does not rank object hypotheses and the object size must be known prior to search. Since illumination changes shift values in the colour histogram bins, histogram intersection is unreliable; therefore Vinod's method is unstable. Schettini's shape-based method deteriorates at low spatial resolution because shape is unreliable. Matas et al.'s graph search process is computationally expensive but can represent 3-dimensional deformable objects with perspective distortions — all of these other algorithms can only represent 2-dimensional objects which are affine distorted.

The proposed algorithm identifies image regions with colours that are similar to the model and uses these as cues. The object size, estimated from the image data, is used to calculate a match measure which rank the cues. This method can represent both 2- and 3-dimensional planar objects in cluttered scenes and is invariant to moderate occlusion.

The remaining Sections of this paper are organised as follows: Section 2 describes the object search methodology adopted; Section 3 details the object search algorithm; Section 4 presents the experimental results and Section 5 the conclusions.

2 Colour Object Search

In earlier work [12], object cues were generated by locating spatially-close regions with colours that were similar to salient model colours. A match measure was then used to determine object presence. Since no region growing was incorporated only part of the object was found; also the match measure used was not robust to occlusion. To improve this method a new algorithm is proposed which simplifies cue generation, incorporates region growing and provides hypotheses ranking with a robust match measure.

In the new algorithm, object search is completed in three stages: cue generation, region growing and hypotheses ranking. In the cue generation phase, image regions with colours that are similar to the colours of the given model are identified; these are cue locations because if the object is present it should be at one of these locations.

The region growing phase is necessary to identify object pixels. Growing begins at the cue region and includes neighbouring pixels with colours which are similar to model colours. To prevent the premature halting of the growing process due to gaps between regions, the image is divided into windows. Growing starts at the window containing the centroid of the cue region and includes windows in the 8-neighbourhood whose pixels increase the object match measure. Growing is terminated when: no more windows contain pixels with colours that are similar to model colours; or the match measure has been maximised; or the object size has been reached. Since the object size was not known a priori, an object size bound is calculated from the cue region's size. The growing process is repeated for different object sizes between the size bound.

Finally, for a given cue a match measure is calculated for each object size. The cue's rank is the maximum match measure and if it exceeds a predefined threshold the object is assumed to be present. The match measure used is colour histogram intersection where the bin colour was defined by the model regions chromaticity coordinates.

Before object search a model database must be generated. For each model colour five parameters are stored: the mean region colour, the minimum and maximum region area percentages for the given colour and the sum of the region area percentages for the given colour. This process is detailed in Algorithm 1.

Algorithm 1: Model Parameter Generation

1. Segment the model image and ignore regions that are smaller than the predefined area threshold.
2. Determine the regions, found in Step 1 with similar colour and calculate the total area of these regions.
3. For each model colour store in the model database: the chromaticity co-ordinates (r, g) of the model colour, the percentage of the total model area (percentage coverage) with the given model colour, and the percentage coverage of the smallest and largest image regions with the given model colour.
4. Repeat Steps 1-3 for each database model.

Database model7 (c.f. Figure 1) is represented by four regions, two yellow and two blue. The largest yellow region is on the right and occupies 72% of the model area while the smallest yellow region (on the far left) occupies 16%. Similarly, the largest and smallest blue regions occupy 10% and 2% of the total model area, respectively. The total percentage coverage for yellow is 88% and blue 12% (c.f. Table 1).

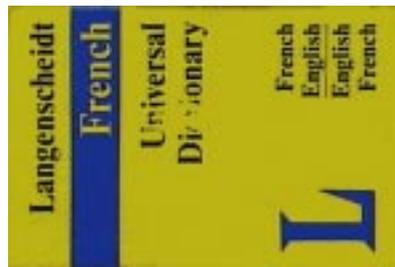


Figure 1: A database model (model7).

The pre-processing requirements for both model and test images are colour constancy and colour image segmentation. Colour constancy is required because the model and test images were not captured under the same illuminant. The colour constancy algorithm used (because it was developed in-house) is due to Hung and Ellis [4]. Image segmentation was achieved using the software colour filter described in Walcott [13]. In this algorithm Khotanzad et al.'s [5] hill climbing clustering algorithm is used to cluster the colour histogram of the image. The pixels belonging to each identified cluster are backprojected into the image and the resulting connected components are treated as regions of constant reflectance.

r	g	Percentage coverage	Percentage coverage of smallest region	Percentage coverage of largest region
0.48	0.46	88.0	16.0	72.0
0.14	0.24	12.0	2.0	10.0

Table 1: The parameters for the model in Figure 1.

3 The Object Search Algorithm

The object search algorithm is detailed in Algorithm 2.

Algorithm 2: Colour Object Search
--

1. Segment the image and remove regions with areas that are less than the minimum area threshold or greater than 50% of the image area. Divide the image into N windows each of size $n \times m$.
2. Repeat Steps 3 - 7 for each database model:
3. **Cue generation:** Cue locations are image regions with colours that are similar to model colours. If (μ_r, μ_g) is a model colour and (μ'_r, μ'_g) an image region colour, then if $C = \sqrt{(\mu_r - \mu'_r)^2 + (\mu_g - \mu'_g)^2} < threshold$ a colour match is recorded.
4. Repeat Steps 5 - 7 for each cue:
5. **Object size determination:** By assuming that the cue region is part of the model and not more than half of it is occluded a minimum (*min_size*) and maximum (*max_size*) object size bound can be calculated:

$$min_size = \frac{number_of_pixels_in_cue_region}{PCLMR} \cdot 100 \quad (2)$$

$$max_size = 2 \cdot \frac{number_of_pixels_in_cue_region}{PCSMR} \cdot 100 \quad (3)$$

where *PCSMR* and *PCLMR* are the percentage coverage of the smallest and largest model regions with a similar colour as the cue region, respectively.

6. **Region Growing:** If s object sizes are used between (and including) *min_size* and *max_size* then the object size increment $k = \frac{max_size - min_size}{s - 1}$.

for (*object_size* = *min_size*; *object_size* <= *max_size*; *object_size* = *object_size* + k)

- (a) *no_object_pixels* = 0;
- (b) Determine the window corresponding to the centroid of the cue region.
- (c) Grow into a neighbouring window (8-neighbourhood) if the colours in that window increases $H(I, M)$, equation (4).
- (d) if $H(I, M)$ increases then *no_object_pixels* = *no_object_pixels* + no of pixels in window with model colours.
- (e) Repeat Steps (c) and (d) until: *no_object_pixels* >= *object_size*, or there are no more neighbouring windows containing model colours; or $H(I, M)$ is maximised.
7. **Match measure:** The maximum $H(I, M)$ for all object sizes is the cue rank. If this value exceeds *match_threshold* then the model is assumed to exist at this cue.

$$H(I, M) = \sum_{j=1}^n \min(I_j, M_j) \quad (4)$$

where $M_j = \frac{\text{number_of_model_pixels_with_colour_}j}{\text{model_size}}$

and $I_j = \frac{\text{number_of_object_pixels_with_colour_}j}{\text{object_size}}$.

In Step 1, any region larger than 50% of the image area is considered a background region — even if this assumption is incorrect the object will still be identified through its other regions. The object size calculation in Step 3 assumes that the cue region is part of the object and not more than half of it is occluded. For example, consider a model with three regions, two red occupying 25% and 50% of the total object area and one green (occupying the remaining 25%). If a red region containing 20 pixels is found

in the image then $\text{min_size} = \frac{20}{50} \cdot 100 = 40$ and $\text{max_size} = 2 \cdot \frac{20}{25} \cdot 100 = 160$ are the

minimum and maximum object sizes in pixels, respectively. There are four important parameters in this algorithm: window size, colour threshold, match percentage threshold; and minimum region area. The window size must be large enough to span gaps between regions due to unclassified pixels; if it is too large, although it speeds up the search, the accuracy of the object location is lost. The colour threshold selected is dependent on the colour constancy algorithm used; the better the algorithm, the smaller the threshold. The match percentage threshold is based on the allowed amount of object occlusion and the minimum region area is selected arbitrarily, however it must be small enough to include the smallest object region that needs to be recognised.

4 Results

The 25 model database used in these experiments is illustrated in Figure 2. The database contains books, cereal boxes, playing cards and a Christmas card box. Many of these models have similar geometry, e.g. models 1, 2, 4, 5, 6, 7 and 15; models 12, 13 and 14; models 20, 21, 22, 23, 24 and 25; and models 9, 10 and 11. The playing card models (9, 10 and 11) have only two colours, white and red and models 10 and 11 have similar colour proportions. Models 13 and 14 have practically the same representative colour regions — other than the text ‘Debugger’ and ‘Assembler.’

The test images used in these experiments are illustrated in Figure 3 (available in colour on the proceedings CDROM). Figure 3(a) contains an occluded model 6 and 14; Figure 3(b) contains model 14. Figure 3(c), (d), (e) and (f) contain model 1 and 2; model 9 and 10 (both occluded); model 2; and model 5, 7, and 12, respectively. These images highlight the performance of the algorithm in conditions such as: cluttered scenes, affine object distortion and object occlusion. Because the model and test images were captured under different illuminants, a colour constancy algorithm [4] was applied to all images before processing.



Figure 2: The model database.

Image	Number of models in image	Correct Match Placement				False positives	Percentage reduction in search space
		1 st	2 nd	3 rd	>3 rd		
(a)	2		2			9	80.7
(b)	1	1				9	78.7
(c)	2	1	1			11	58.3
(d)	2	2				5	0.0
(e)	1		1			9	60.1
(f)	3	1	1	1		8	41.6

Table 2: A summary of the results of applying Algorithm 2 to the images of Figure 3.

Table 2 presents a summary of the results of applying Algorithm 2 to the six images of Figure 3. The first column of this table contains the image identifier (Figure 3(a) - (f)), the second column contains the number of database models that are present in the given image. The third column contains the placement of each match, that is whether the cue with the best match value (1st) represents the object, or is the second best (2nd), or the third (3rd) best or worse (>3rd). The fourth column contains the total number of false positives that have occurred, and finally column five gives the percentage

reduction in the search space (which is defined as the number of image windows containing the localised model (and the false positives) over the total number of image windows).



Figure 3: The images used in the object search experiments.

In Figure 3(a) both models 6 and 14 had the second best rank at their correct image location, however there were 9 false positives. The percentage reduction in the search space was 80.7% and the percentage reduction of the models present in the image is 44%. The remainder of Table 2 is interpreted in the same way. It is important to note however that there was no appreciable reduction in the search space for Figure 3(d).

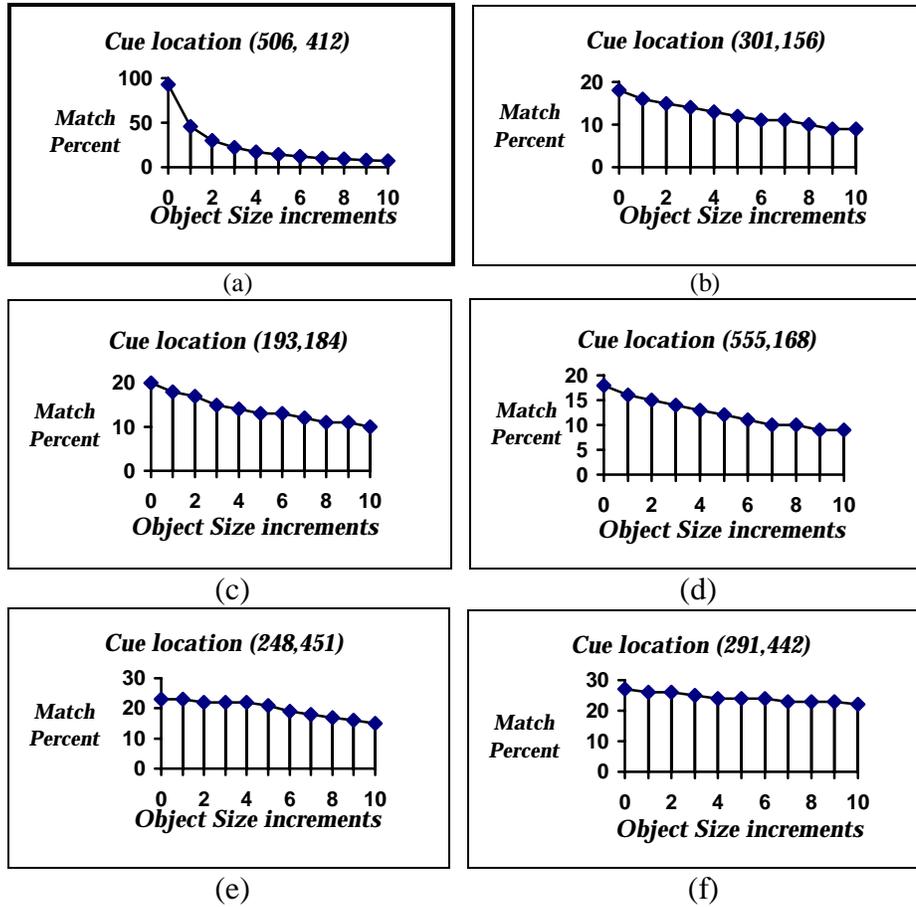


Figure 4: The match percentages at increasing object size increments (at the cue locations indicated) when searching for model6 in image Figure 3(a). In this example the location which yields a match percentage of 93%, (a) contains model6.

To illustrate the search process, consider the search for model6 in Figure 3(a). The parameters used in all the experiments were: window size 10×10 pixels, minimum object match percentage 89.5% and 11 object sizes were selected between *min_size* and *max_size* (all equally spaced) for each region cue found. In Step 3 of Algorithm 2, six cue regions were identified with centroids at (x,y) co-ordinates: (506,412), (301,156), (193,184), (555,168), (248,451), and (291,442). At each cue, the growing process was initiated for each of the 11 object sizes (c.f. Figure 4) and the match measure calculated. The only cue which generated a match percentage greater than 89.5% is at co-ordinates (506,412) (c.f. Figure 4(a) and 5(a)) where a match percentage of 93% was calculated for object size increment 0 (i.e. *min_size*); therefore this is the solution.

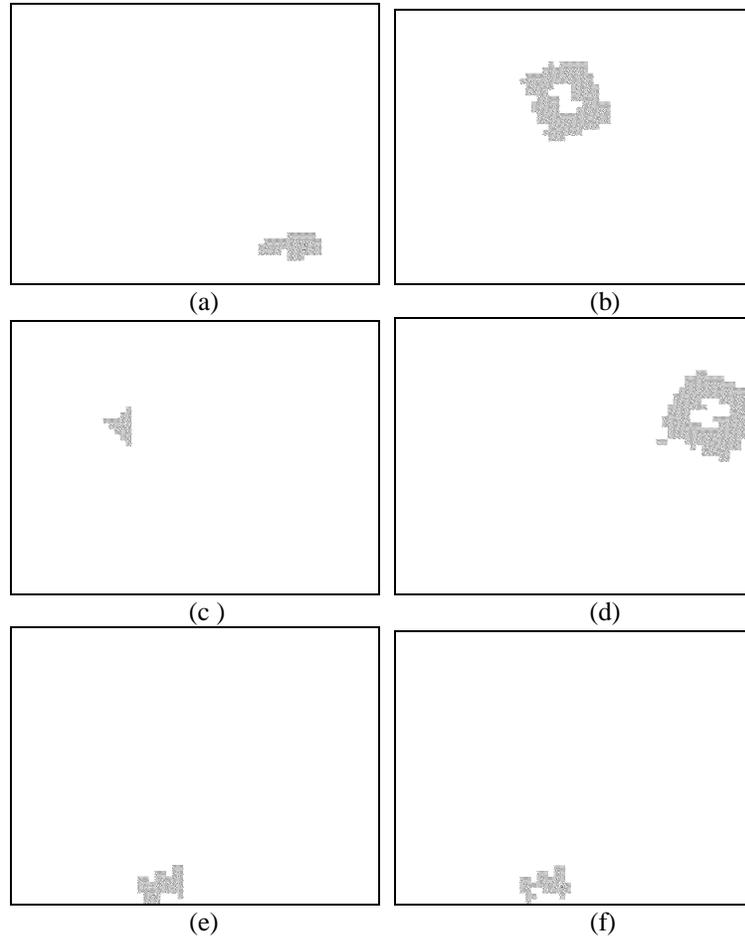


Figure 5: The six cue regions.

5 Conclusions

In this paper a colour object search algorithm capable of locating 2- or 3-dimensional planar objects was presented. Successful searches were performed in: complex, cluttered scenes; high and low spatial resolutions of the object; affine object distortion; and up to 50% occlusion of the object area. The recognition rate for these experiments was 45% at both rank 1 and 2 and 10% at rank 3. There were no false negative results but 51 false positives. The average reduction in the model database and the object search space were 68% and 53%, respectively.

This algorithm outperforms all of the colour object localisation/search algorithms presented in this paper [8] [9] [10] [11] [12] [15] except Matas et al.'s colour adjacency graph [6]; these methods are 2-dimensional planar (except Matas et al. which is 2- or 3-dimensional) while the adopted method is 2- or 3-dimensional planar. The colour histogram intersection metric used by this algorithm is more robust to occlusion than Wixson and Ballard's [15] and our [12] colour ratio measure; and more stable under changes in illumination and the position of the light source than Vinod [11] (since

shifts in histogram bin values disrupt the metric) because a given model colour falls into only one bin.— also, the normalised colour space is independent of image geometry. In histogram backprojection determining peaks in the confidence space of complex scenes with several false positives, is non-trivial. In addition, in the adopted algorithm object size is calculated from image data and fewer floating point numbers are required to represent the model (5 numbers per model colour).

The algorithm is incapable of representing perspectively distorted 3-dimensional objects or objects with similar colour proportions but different topologies (whereas Matas et al. can). Possible improvements include the calculation of a more accurate object size (for example based on the area of a non-occluded region) and the improvement of the cue generation process by using for example pairs of related regions (e.g. adjacent).

References

- [1] Grimson, W., E., L., "*Object Recognition by Computer: The Role of Geometric Constraints*", MIT Press, Cambridge, Massachusetts, London, England, 1990.
- [2] Healey, G., E., Binford, T., O., "The Role and Use of Colour in a General Vision System", *proc. DARPA IV Workshop*, USC, CA, USA, 1987, pp 599-613.
- [3] Hilbert, D., R., "*Color and Color Perception: A Study in Anthropocentric Realism*", Center for the Study of Language and Information (CSLI), 1986.
- [4] Hung, T, W, R, Ellis, T, "Spectral Adaptation with Uncertainty using Matching", *IEE Proc. of the 5th Int. Conf. on Image Processing and its Applications*, Scotland, 1995, pp 786-790.
- [5] Khotanzad, A, Bouarfa, A, "Image Segmentation by a Parallel, Non-Parametric Histogram Based Clustering Algorithm", *Pattern Recognition*, **23**, 9, 1990, pp 961-973.
- [6] Matas, J, Marik, R, Kittler, J, "On Representation and Matching of Multi-coloured Objects", *Proc. of ICCV*, Boston, 1995, pp 726-732.
- [7] Mundy, J., L., Zisserman, A., "*Geometric Invariance in Computer Vision*", MIT Press, Cambridge, Ma, London, 1992.
- [8] Schettini, R., "Multicolored Object Recognition and Location", *Pattern Recognition Letters*, **15**, 1994, pp 1089-1097.
- [9] Stricker, M., Orengo, M., "Similarity of Color Images", in *Stor. and Retrieval for Image and Video Databases III, SPIE Proc. Series*, **2402**, Feb. 1995, pp 381-392.
- [10] Swain, M., J., Ballard, D., H., "Indexing via Color Histograms", *Proc. ICCV*, 1990, pp 390-393.
- [11] Vinod, V., V., Murase, H., "Object Location Using Complementary Color Features: Histogram and DCT", in *Proc. of ICPR 96*, 1996, pp 554-559.
- [12] Walcott, P, Ellis, T, "The Localisation of Objects in Real World Scenes Using Colour", *Proc. of 2nd ACCV.*, Singapore, December 1995, pp 243-247.
- [13] Walcott, P, "Object Recognition Using Colour, Shape and Affine Invariant Ratios", *BMVC96*, Edinburgh, Scotland, September 1996, pp 273-282.
- [14] Walcott, P., "*Colour Object Search*", PhD thesis, City University, July, 1998.
- [15] Wixson, L, Ballard, D, "Real-time Detection of Multi-coloured Objects", *SPIE Sensor Fusion II: Human and Mach. Strategies*, **1198**, Nov., 1989, pp 435-446.