

UNIVERSITY OF WESTMINSTER



**WestminsterResearch**

<http://www.wmin.ac.uk/westminsterresearch>

## **Generic colour image segmentation via multi-stage region merging.**

**Gaurav Gupta**  
**Alexandra Psarrou**  
**Anastassia Angelopoulou**

School of Electronics and Computer Science

Copyright © [2009] IEEE. Reprinted from the proceedings of the 10th Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS '09). IEEE, pp. 185-188. ISBN 9781424436095.

This material is posted here with permission of the IEEE. Such permission of the IEEE does not in any way imply IEEE endorsement of any of the University of Westminster's products or services. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE. By choosing to view this document, you agree to all provisions of the copyright laws protecting it.

---

The WestminsterResearch online digital archive at the University of Westminster aims to make the research output of the University available to a wider audience. Copyright and Moral Rights remain with the authors and/or copyright owners. Users are permitted to download and/or print one copy for non-commercial private study or research. Further distribution and any use of material from within this archive for profit-making enterprises or for commercial gain is strictly forbidden.

---

Whilst further distribution of specific materials from within this archive is forbidden, you may freely distribute the URL of the University of Westminster Eprints (<http://www.wmin.ac.uk/westminsterresearch>).

In case of abuse or copyright appearing without permission e-mail [wattsn@wmin.ac.uk](mailto:wattsn@wmin.ac.uk).

# GENERIC COLOUR IMAGE SEGMENTATION VIA MULTI-STAGE REGION MERGING

*Gaurav Gupta, Alexandra Psarrou, Anastasia Angelopoulou*

Harrow School of Computer Science, University of Westminster, Middlesex HA1 3TP, UK

## ABSTRACT

We present a non-parametric unsupervised colour image segmentation system that is fast and retains significant perceptual correspondence with the input data. The method uses a region merging approach based on statistics of growing local structures. A two-stage algorithm is employed during which neighbouring regions of homogeneity are traced using feature gradients between groups of pixels, thus giving priority to topological relations. The system finds spatially cohesive and globally salient image regions usually without losing smaller localised areas of high saliency. Unoptimised implementations of the method work nearly in real-time, handling multiple frames a second. The system is successfully applied to problems such as object detection and tracking.

## 1. INTRODUCTION

Image segmentation is commonly defined as the identification of homogeneous regions within an image. The segmentation is then guided by the interpretation of homogeneity, usually involving colour or spatial distribution or both. Popular approaches include region-based methods, edge-based methods, hybrid techniques incorporating both regions and edges, histogram-based methods, and graph-based methods.

Region-based methods [1, 2, 3] group pixels into segments based on some pixel similarity measure and threshold values to indicate whether the similarity test is passed. Edge-based methods [4, 5], on the other hand, find region boundaries by applying edge detection mechanisms, and are limited due to the high number of edges found and by the need to have an effective mechanism to close edges and form contained regions. Histogram-based methods [6, 7] analyse peaks in dominant colours in order to establish cluster centers to which pixels are assigned. Hybrid methods [8, 9] use both regions and edges but require complex mechanisms to draw correspondences between the two. Graph methods [10, 11] represent pixels as nodes on a graph and pixel groupings as links between nodes. Graph-based methods are usually computationally complex due to the huge set of potential pixel relations.

Segmentation methods strive to achieve a balance between the resolution of the results and the generic applicability of the method. At one end of this spectrum lie methods that extract the primary salient regions of images. For instance,

a salient region segmentation algorithm [12] finds broad regions that are most likely to capture human attention. The limitation of this is that the details contained in these broad regions are lost. Conversely, methods such as [10, 3] produce greater detail but the perceptual significance of each of their segments grows less obvious since objects that may be considered whole segments by humans may be split into multiple regions by the program.

The system we propose aims for a set of regions that reflects regions of primary saliency in the image, but which also retains a level of detail where the visual importance of localised zones is high. Our system is region-based for reasons of low complexity and the need to prioritise topological proximity. The system works in two stages, the first quickly establishing class labels from neighbouring pixel colour distances, and the second merging the preliminary set of classes or segments. Since we consider both colour and segment size in the second merging stage, fewer and larger segmented regions would result, unless extremely significant local features force the separation of smaller areas.

The paper is organized as follows. Section 2 explains the framework of the segmentation mechanism. Section 3 gives a set of results we obtained using a range of images of various types and complexity. Section 4 concludes with a summary of the presented work and identifies future directions of work.

## 2. SEGMENTATION STRATEGY

The region merging strategy consists of two broad stages: a preliminary pixel-level class label assignment stage, followed by an iterative class merging stage.

The merging procedure is dependent on two threshold values, the pixel merging distance  $d_p$  and the segment merging distance  $d_s$ . Experimental tuning sets these parameters to  $d_p = 10.5$  and  $d_s = d_p * r$ , where the segmentation factor (inversely proportional to the resolution of the segmentation)  $r = 15 * 10^6$  provides a good segmentation of large salient regions while maintaining some smaller regions of high importance.

In stage 1, we carry out a preliminary class label assignment for each pixel based on a threshold value between immediate neighbours. We start with an empty set  $S$  of class labels, and move through the entire image row-by-row from top left to bottom right, using a label assignment strategy to

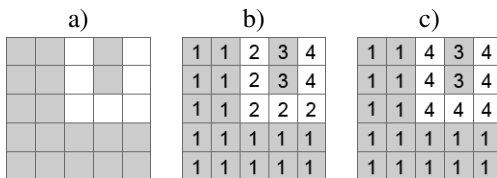
populate  $S$  with possible class labels  $s_i$ , where  $i$  is the label counter. On the very first pixel  $P_{m,n}$ ,  $m = 1, n = 1$ , the label counter  $i$  is set to 1 and a new element  $s_1$  is inserted into  $S$ . Thus  $P_{1,1}$  is assigned the class label  $s_1$ . We then carry out the following steps until there are no more image pixels to process:

1. Try to assume a neighbouring pixel label using the procedure *Seek*. If this succeeds, move on to the next pixel.
2. If it fails, increment the label counter  $i$  and assign the new label  $s_i$  to the current pixel, inserting this label into the set of labels  $S$ .

A  $k \times l$  kernel window  $K$ , where  $k \bmod 2 \neq 0, l \bmod 2 \neq 0$  and  $k > 1, l > 1$ , is employed at several stages of the segmentation. We keep the window at the smallest possible non-trivial size, which is  $k = 3, l = 3$ . When  $K$  is positioned over pixel  $P_{m,n}$  of image  $I$ , the kernel window coordinates are represented by  $K_{x,y}$ .

*Seek* procedure: We center the kernel window  $K$  over the currently considered pixel and proceed to compare the feature distance,  $dM$  (the Manhattan distance) between the center pixel and other pixels of the kernel that lie within the image region. Since our kernel is  $3 \times 3$ , we have eight possible neighbours for each center pixel considered, and thus eight neighbour distances  $dM_c, c = 1 : 8$ . For each of these eight, the class label corresponding to the smallest  $dM_c$  that falls within the allowable pixel merging threshold  $d_p$  is assigned to the center pixel. If none of the  $dM_c$  values that satisfy the threshold already possess a class label then this procedure fails.

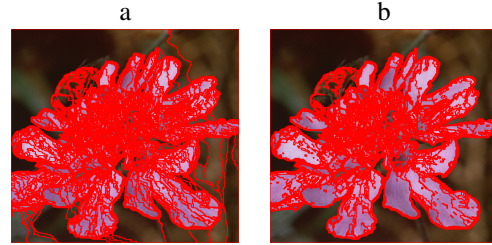
By the end of this process, we get a label map for all the pixels in the image. However, there remains a problem. Since we proceed from left to right, top to bottom, there are cases when the class labelling splits what should be a single segment into multiple classes. Figure 1 shows a simple example where the *Seek* procedure assigns class labels 2 and 4 to pixels actually belonging to a single class. The *Seek'* procedure corrects this and the label 2 is dropped from the set of labels, the entire segment now being assigned the label 4.



**Fig. 1.** Label assignment: a) Image, b) *Seek*, c) *Seek'*

Using *Seek'*, we run through all the pixels on a second pass, this time merging segments indicated by neighbouring pixels satisfying the pixel merging threshold but with different class labels. Although this step could have been avoided

by making the first step more complex, we make significant performance gains by having two low complexity passes instead of a single complex pass. Figure 2 shows the *Seek'* pass correcting the initial label assignments established by the *Seek* pass.



**Fig. 2.** Label correction between *Seek* (a) and *Seek'* (b)

*Seek'* procedure: We again center the kernel window  $K$  over the currently considered pixel and proceed to compare the feature distance,  $dM$  between the center pixel and other pixels of the kernel that lie within the image region. From the eight neighbour distances  $dM_c, c = 1 : 8$ , we eliminate those that correspond to pixels with different class labels from the center pixel and get an updated set of neighbour distances  $dM'_c$ . Now considering this set, the class label corresponding to the smallest  $dM'_c$  that falls within the allowable pixel merging threshold  $d_p$  is assigned to the center pixel, using the *Merge* procedure. If none of the  $dM'_c$  values that satisfy the threshold already possess a class label then this procedure fails.

*Merge* procedure: When merging two pixels or pixel regions with classes  $u$  and  $v$ , the pixels belonging to both classes are all set to  $u$ . Thus the two segments previously identified by labels  $u$  and  $v$  are now identified by a single label  $u$ .

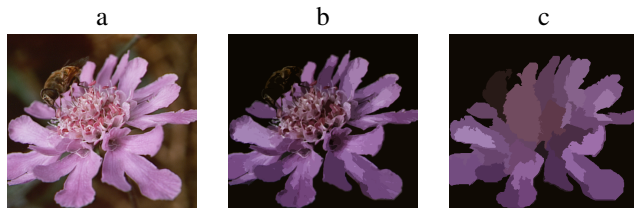
Next we move to stage 2 of the segmentation, which is similar to stage 1, with three differences that redefine the functionality of the *Seek'* procedure to get a new procedure *Refine*. First, we now only consider the kernel window and pixel level information in order to identify neighbouring segments. Neighbours are now defined as, for kernel  $K$  centered at image coordinates  $P_{m,n}$ , segments for which  $K$  contains pixels from each segment and at least one of the segments places a pixel at the kernel center,  $K_{((m+1)/2, (n+1)/2)}$ . Secondly, the distance measure  $dM$  is now between neighbouring segments instead of neighbouring pixels. The feature vector for each segment is recomputed as the mean of the individual feature vectors of their component pixels. The third difference is in the merging threshold, which is now  $d_s$  instead of  $d_p$ . This threshold  $d_s$  is further magnified by the sizes of the particular segments being considered for the merge. The modified procedure is as follows:

*Refine* procedure: For kernel  $K$  at  $P_{m,n}$ , if segment label  $s_U$  at its center  $K_{((m+1)/2, (n+1)/2)}$  differs from that at another  $K_{(m,n)}$ , then  $s_U$  and  $s_V$  are neighbouring segments.

For each pair of neighbouring segments identified by an instance of  $K$ , we get a set of distance measures  $dM_c$ . The distance measures are calculated as the distance between the means of the feature vectors of all the pixels belonging to each segment. Each distance measure is further multiplied by the number of pixels in each of the two segments being considered, i.e. for segments  $s_U$  and  $s_V$  with number of pixels  $N_a$  and  $N_b$  respectively,  $dM_c = dM_c * N_a * N_b$ . The segment corresponding to the smallest  $dM_c$ , and which falls within the allowable segment merging threshold  $d_s$ , is merged using procedure *Merge* with the segment identified by the class label at the center of  $K$ . If none of the  $dM_c$  values satisfy the threshold then this procedure fails.

Thus, starting at the top left of the image and proceeding row-by-row to the bottom right, we iterate through the following steps until in any single run through the entire image no segment merges occur:

1. For the current segment, try to assume a neighbouring segment label using the new procedure *Refine*, modified in the three ways as described above. Move to the next pixel if it *fails*.
2. If it succeeds, recompute properties global to the newly merged segment and its feature vector as the mean of the feature vectors of its component pixels. Move to the next pixel.



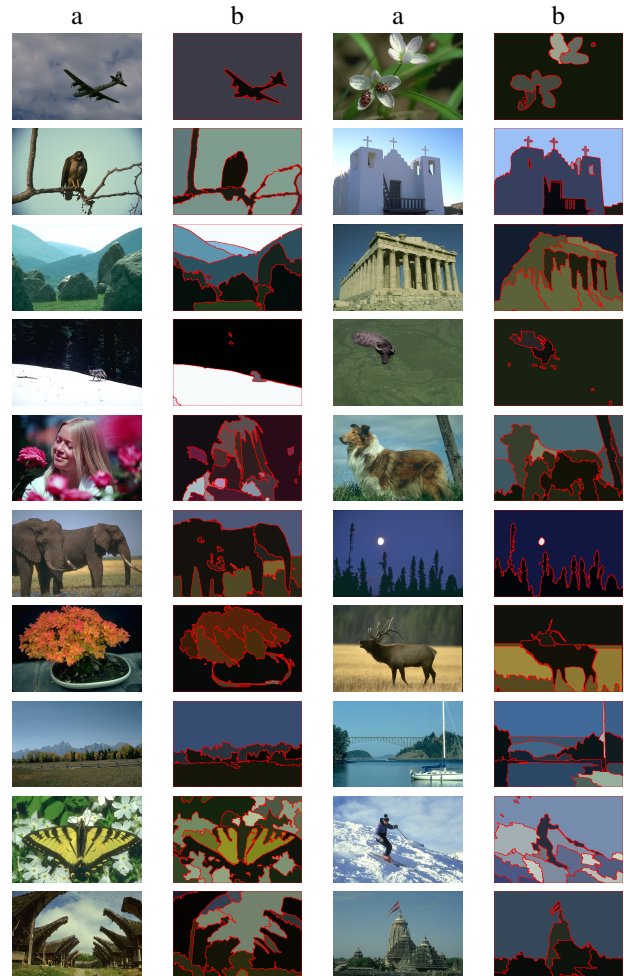
**Fig. 3.** Segmentation stages: a) Original image, b) Over-segmented results after Stage-1 *Seek'*, c) Final results after Stage-2 processing

The segmentation is now complete. Figure 3 shows the two-stage results. For  $r = 15 * 10^6$ , we typically get within ten and twenty final segments. Note: figure 3-b represents the same processing stage as figure 2-b.

### 3. RESULTS

Figures 4 and 5 show the segmentation results on a variety of images from the Berkeley Segmentation Data Set (BSDS) [13]. The results are seen to be good across a variety of images. While the system was designed to handle live video images, and thus tuned to handle camera noise and illumination issues, due to limitations of paper length we only present results from the commonly cited BSDS.

The segmentation system is computationally efficient. Our unoptimised implementation on a dual core 2.40-GHz 1.98-GB RAM Windows XP machine produces speeds between 0.2 and 1 seconds a frame for the BSDS  $481 * 321$  image size, depending on image complexity.



**Fig. 4.** Images from the Berkeley Segmentation Data Set. a) Original, b) Segmented

In our system, segments represent zones of interest, and regions where multiple segments are concentrated represent possible points of gaze fixation. Images in which there are fewer and weaker dominant points of fixation are indicated by a segmentation set consisting of fewer and larger regions, similar to the human perception taking a little longer to identify something significant to look at in the image.

As seen from the results, we sacrifice uniformly spread local detail in order to gain global saliency zones. The system keeps intact localised small segments only where it finds the visual importance of the segment to be very high. We note however that once broad regions of interest are identified, it is possible to apply the same system at higher resolutions to pick up greater detail from those regions.

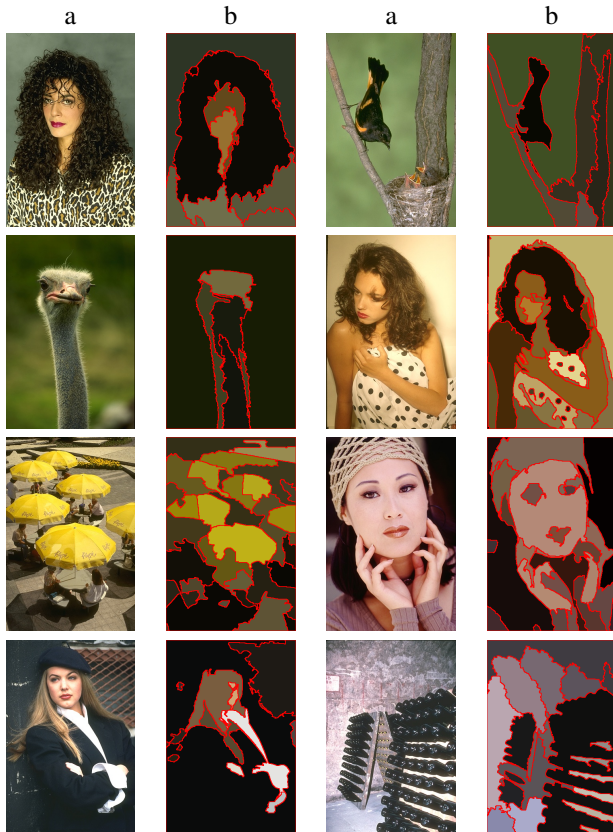


Fig. 5. More images from BSDS. a) Original, b) Segmented

#### 4. DISCUSSION AND FUTURE WORK

We have presented a natural colour image segmentation method that is fast and maintains a level of perceptual correlation with the input. The method produces segments that signify salient image regions, while retaining smaller intra-object regions should they present sufficiently salient features. Simple experiments using the segmentation system to track a ball were successful which validates the property of perceptual coherence of this system, although we lack the scope in this paper to present more of the details.

Future work will explore more complex feature statistics for region merging, such as boundary statistics, neighbouring segment features, and textures.

#### 5. REFERENCES

[1] S. Y. Wan and W. E. Higgins, "Symmetric region growing," *IEEE Transactions on Image Processing*, vol. 12, no. 9, pp. 1007–1015, 2003.

[2] S. C. Cheng, "Region-growing approach to colour segmentation using 3-d clustering and relaxation labeling," *IEEE Proceedings - Vision, Image, and Signal Processing*, pp. 270–276, 2003.

[3] L. Garcia-Ugarriza, E. Saber, V. Amuso, M. Shaw, and R. Bhaskar, "Automatic color image segmentation by dynamic region growth and multimodal merging of color and texture information," *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2008*, pp. 961–964, 2008.

[4] W. Y. Ma and B. S. Manjunath, "Edge flow: a technique for boundary detection and image segmentation," *IEEE Transactions on Image Processing*, vol. 9, no. 8, pp. 1375–1388, 2000.

[5] G. Iannizzotto and L. Vita, "Fast and accurate edge-based segmentation with no contour smoothing in 2-d real image," *IEEE Transactions on Image Processing*, vol. 9, no. 7, pp. 1232–1237, 2000.

[6] H. D. Cheng, X. H. Jiang, and J. Wang, "Color image segmentation based on homogram thresholding and region merging," *Pattern Recognition*, vol. 35, pp. 373–393, 2002.

[7] K. S. Chenaoua, A. Bouridane, and F. Kurugollu, "Unsupervised histogram based color image segmentation," *10th IEEE International Conference on Electronics, Circuits and Systems*, vol. 1, pp. 240–243, 2003.

[8] J. Fan, D. K. Y. Yau, A. K. Elmagarmid, and W. G. Aref, "Automatic image segmentation by integrating color-edge extraction and seeded region growing," *IEEE Transactions on Image Processing*, vol. 10, no. 10, pp. 1454–1466, 2001.

[9] T. Gevers, "Adaptive image segmentation by combining photometric invariant region and edge information," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 6, pp. 848–852, 2002.

[10] W. Tao, H. Jin, and Y. Zhang, "Color image segmentation based on mean shift and normalized cuts," *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics*, vol. 37, no. 5, pp. 1382–1389, 2007.

[11] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *International Journal of Computer Vision*, vol. 59, pp. 167–181, 2004.

[12] Y. Kuan, C. Kuo, and N. Yang, "Color-based image salient region segmentation using novel region merging strategy," *IEEE Transactions on Multimedia*, vol. 10, no. 5, pp. 832–844, 2008.

[13] D. Martin, Ch. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," *Proc. ICCV*, pp. 416–425, 2001.