

# Image Segmentation Using Dynamic Run-Length Coding Technique

William O. McCallister<sup>1</sup> and Chih-Cheng Hung<sup>2</sup>

<sup>1</sup>Lockheed Martin Aeronautics Company, 86 S. Cobb Dr. Mail Zone 0033, Marietta, GA 30063, USA

William.o.mccallister@lmco.com

<sup>2</sup>School of Computing and Software Engineering,  
Southern Polytechnic State University, Marietta, GA 30060 USA  
chung@spsu.edu

**Abstract.** *In this study, a new segmentation algorithm based on a modified Dynamic Window-based gray-level Run-Length Coding (DW-RLC) applied to neighboring pixels is proposed. The method is applied to gray scale images, RGB color images, and images in the HLS color space. The proposed algorithm, which has a fast image segmentation performance from the experiments, can be categorized as a region growing method. Experimental results show that the proposed algorithm has several advantages in segmenting images.*

## 1 Introduction

Image segmentation is an important step in many computer vision and image processing applications. In a general definition, image segmentation is the partition of an image into several nonoverlapping regions. Image segmentation is an elusive goal due to varying statistical characteristics such as sensitivity to noise, reflectance property of object surfaces, lighting environments, and so on [1].

Image segmentation using contextual information has been studied for a while. Context is important in image segmentation because contiguous pixels are likely to belong to the same region. ECHO [2] is an example of an image segmentation algorithm that uses the region growing technique to segment and classify an image by applying the maximum likelihood classifier. Markov Random Fields (MRF) model is widely use as a convenient means for introducing context, or dependence among neighboring pixels [3].

Pixel similarity and interpixel correlation has been used as one of the properties in image compression. In this study, a new segmentation algorithm based on a modified gray-level run-length coding is proposed. The algorithm is based on similarity of pixel values and spatial connectivity, described in sections 2 and 3. In other words, spatial connectivity and pixel similarity are used as a means introducing the contextual information in the proposed segmentation algorithm. The new algorithm is described in section 4. The algorithm is extended to multi-band and color images in section 5. Experimental results are described and illustrated in section 6. The conclusion then follows.

## 2 Gray-Level Run-Length Coding and Spatial Connectivity

The basic Run-length coding (RLC) is an image compression method that works by counting the number of adjacent pixels along a scan line with the same gray-level value [4] in a binary image. This count, called the run-length, is then coded and stored.

The RLC technique can be applied to gray-level images by using bit-plane coding. A more sophisticated RLC algorithm for encoding gray-level images is the dynamic window-based RLC (DW-RLC) [4]. Unlike the basic RLC technique, this algorithm does not use the criterion of the runs being the same value. Instead the algorithm allows for the runs to fall within a gray-level range, which is called the dynamic window range. This range is dynamic because it starts out with any range, and maximum and minimum values are narrowed down to the actual range as each pixel value is encountered. This procedure continues until a pixel is found out of the current range limits. The value recorded for this run will be the average of the included pixels. Then, a new run will be started and the similar procedure will be repeated. Details of this procedure are given in Section 4 (the direction is not limited to the East).

Connectivity describes the spatial relationship between one pixel and other contiguous pixels of the same class. If a pixel is connected to a very large object, or an object of a particular shape, it is most likely coming from this particular class [5,6]. Connectivity ensures that all the pixels within a certain range are not randomly distributed (noise).

## 3 The Proposed Segmentation Algorithm

The proposed algorithm includes pixels in a segment based on spatial connectivity and similarity of pixel parameters. As the RLC algorithm only utilizes horizontal spatial information for the image compression, a new algorithm is proposed based on the neighboring spatial information concept. In this new segmentation algorithm, the spatial information is not limited to the horizontal direction; all the neighboring spatial information will be incorporated in the segmentation process. The neighboring spatial information is defined based on the 4-connectivity or 8-connectivity. The proposed segmentation algorithm is sketched below:

### 3.1 Algorithm Description

Step 1: Define the window range that pixel values may have in a segment,  $R$ . Note: implementations using integer pixel values will define  $R = \text{desired window range} - 1$ .

Step 2: Select a starting pixel.

Step 3: Calculate the initial maximum (MAX) and initial minimum (MIN) of the window range for the starting pixel value,  $PV$ , being considered, respectively:

$$\text{MIN} = PV - R \quad (1)$$

$$\text{MAX} = \text{PV} + \text{R} \quad (2)$$

Step 4: Select a new candidate pixel with connectivity to the current segment. Note that the location of this next pixel depends on the connectivity used (4 or 8 as defined in section 3).

Step 5: Calculate the new maximum (NMAX) and new minimum (NMIN) of the window range for the candidate pixel value, NPV, being considered.

$$\text{NMIN} = \text{NPV} - \text{R} \quad (3)$$

$$\text{NMAX} = \text{NPV} + \text{R} \quad (4)$$

Step 6: If the candidate pixel value is in the range of [MIN, MAX], narrow down the range:

if  $\text{NMIN} > \text{MIN}$  then  $\text{MIN} = \text{NMIN}$ .

if  $\text{NMAX} < \text{MAX}$  then  $\text{MAX} = \text{NMAX}$ .

Hence, the new (dynamic) window range would be [MIN, MAX]. Repeat steps 4-6 for the next pixel if the candidate pixel value is in the range of [MIN, MAX]. Otherwise, go to step 7.

Step 7: If the candidate pixel value is not in the range of [MIN, MAX], average all the pixels scanned so far (this average will be the segment's value) and go to step 2 to start a new segment.

Step 8: Exit whenever each pixel in the image is examined.

### 3.2 Seed Fill

In the algorithm described above, step 4 requires a way to visit all of the connected pixels in a region. One of the many possible candidates will be chosen for the next pixel. This was implemented with a seed fill algorithm [7,8]. Seed fill algorithms start with a single pixel, the seed, and examine connected pixels to determine if they belong in the region. Many seed fill algorithms use a stack to record the location of accepted pixels, so that each can eventually have its neighbor pixels checked. With this type of algorithm, step 4 is performed by popping a pixel location off the stack, and selecting one of its neighbors as a candidate. Subsequent executions of step 4 will select other neighbors, until all have been considered. The next time through step 4, a new pixel location is popped, and one of its neighbors selected.

### 3.3 Seed Pixel Selection

Segmentation results are sometimes influenced by the location of seed pixels. The segmentation algorithm selects the specific range of pixel values that will be included in a segment from the first pixels encountered in the segment. As new pixel values are encountered, the range of acceptable values will converge on the desired range. Thus the first few pixels influence which subsequent pixels will be selected. Pixels on the edge of a segment may be a blend of pixel values properly associated with the segment, and values of other segments immediately adjacent. It is therefore desirable

that seed pixels not be at the edge of a segment. To avoid this, random seed pixels are chosen.

## 4 Color Image Segmentation

The algorithm described above can be applied directly to single band or gray scale images. To apply the algorithm to multiple band or color images, each band or color channel is assigned a range, and minimum and maximum values are tracked for each channel as new pixels are encountered. To accept a pixel as part of the current segment, all channels of the pixel must be within their respective minimum and maximum limits.

### 4.1 Color Space for Segmentation

The segmentation algorithm can exploit the information about an image that is directly provided by its representation in a particular color space. For example, if there is a hypothesis that images of interest should be segmented into regions of similar hue and saturation, a color space such as HLS could be used. In that case, relatively narrow ranges could be specified for hue and saturation channels, while lightness would be allowed a wider range to accommodate variations in illumination in the original scene.

Hue, lightness and saturation are treated as signal bands, so when a pixel is found with h, l or s values out of their ranges, the segment is completed, as in step 7 above.

HLS color space with range parameters as described above has been a successful strategy for this algorithm with the test images that were used. Images were converted from RGB to HLS according to the algorithm given in [8]. H, L and S channels were each assigned a range and tracked with minimum and maximum limits. As noted in [9], hue is represented around a closed hexagon, so the minimum and maximum hue parameter values are continuous. A hue range that extends below the minimum hue value, or above the maximum, must include adjacent hue values at the opposite end of hue parameter values.

RGB color space has also been used with the algorithm. In that case, a single range parameter can be used for red, green and blue channels.

## 5 Experimental Results

Experiments on several images are shown in Figures 1 - 4. It should be noted that we did not apply any pre-processing technique to the original images to remove any noisy signals. The 4-connectivity for spatial information was used in our experiments for the images shown in this paper. The seed fill algorithm will always access the corner pixels eventually, hence there has been no benefit to an 8-connected fill algorithm. Figures 1 and 2 illustrate the effect on segmentation in HLS space by varying the lightness range. Figure 2 left shows most of the aircraft in two segments. Reduc-

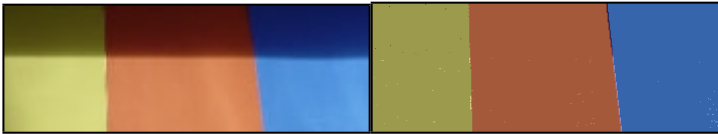
ing the lightness range results in a greater number of segments as shown in Figure 2 right. Figure 3 illustrates the algorithm's ability to track a segment through a wide variation in original subject illumination. Figure 4 illustrates the algorithm's performance on a low contrast image. In this case, RGB color space was used successfully. Table 1 gives segmentation time for different sizes of two images.



**Fig. 1.**An original image.



**Fig. 2.** Left: A segmented image of Figure 1 in two segments, hue range 40 from 0..360, lightness range 0.7 from 0..1, saturation range 0.7 from 0..1; right: four segments, same ranges as used in the left except lightness range 0.3.



**Fig. 3** Left: Colored paper photographed with shadow; right: segmentation of the left image. HLS segmentation ranges same as figure 2 left.



**Fig. 4.** Left: Low contrast highly textured original image; right: RGB segmentation of the left image. R, G and B range 75 from 0..255.

**Table 1.** The time complexity for six different sizes of images, measured in milliseconds in a 850 MHZ Intel processor.

Images	Image Size (pixels)	Time for Segmentation (Milliseconds)
Forest	1225 x 1838	2071
“	612 x 918	521
“	306 x 459	140
Naboo	504 x 507	201
“	252 x 253	47
“	126 x 126	8

## 7 Conclusion and Future Work

A new segmentation algorithm is proposed based on the neighboring spatial information concept, which is derived from the dynamic window-based run-length coding (DW-RLC) algorithm. The proposed segmentation algorithm can be categorized as a region growing method. In this new segmentation algorithm, the spatial information is not limited to the one-way horizontal direction as done in run-length coding; all the neighboring spatial information was incorporated in the segmentation process. The neighboring spatial information is defined based on the 4-connectivity or 8-connectivity. Experimental results show that the proposed algorithm is very promising and worth further exploration. A comparison with other segmentation algorithms is underway.

## References

1. Chang I. and Park, R., "Segmentation Based on Fusion of Range and Intensity Images Using Robust Trimmed Methods," Pattern Recognition, Vol. 34, No. 10, pp. 1951-1962, 2001.
2. Kettig, R. L. and Landgrebe, D. A., "Classification of multispectral image data by extraction and classification of homogeneous objects", IEEE Trans. on Geoscience Electronics, Vol. GE-4, No. 1, January, pp.19-26, 1976.
3. Dubes, R. C., Jain, A. K., Nadabar, S. G. and Chen, C. C. , "MRF Model-Based Algorithms For Image Segmentation," IEEE International Conference (CH289805/90), pp. 808-814, 1990.
4. Umbaugh, S. B., Computer Vision and Image Processing: A Practical Approach Using CVIPools, Prentice Hall PTR, 1998.
6. Kittler, J. and Foglein, J., "Contextual classification of multispectral pixel data," Image and Vision Computing, Vol 2, No 1, 13 – 29, 1984.
6. Gurney, C. M. and Townshend, J. R. G., "The use of contextual information in the classification of remotely sensed data", Photogrammetric Engineering and Remote Sensing, Vol. 49, No. 1, 55-64, 1983.
7. Rogers, D. F., "Procedural Elements For Computer Graphics, McGraw-Hill, pp. 85-88, 1985.
8. Foley, J. D., vanDam, A., Feiner, S. K., and Hughes, J. F., "Computer Graphics Principles and Practice", Addison-Wesley, 1990.
9. Zhang, C. and P. Wang, P., "A new method of color image segmentation based on intensity and hue clustering", International Conference on Pattern Recognition, Vol 3, 2000.