# Image Segmentation Using Grey Scale Weighted Average Method and Type-2 Fuzzy Logic Systems

Saikat Maity[1] and Jaya Sil[2]

[1] Dr. B.C. Roy Engineering College
Tel.: +919932252306
saikat.maity@bcrec.org
[2] Bengal Engineering & Science University (B.E.S.U)
Tel.: +919433283641
js@cs.becs.ac.in

**Abstract.** In the paper, the difficulty in image segmentation based on the popular level set framework to handle an arbitrary number of regions has been addressed. There is very few work reported on optimized segmentation with respect to the number of regions. In the proposed model, first the image is classified using type-2 fuzzy logic to handle uncertainty in determining pixels in different color regions. Grey scale average (GSA) method has been applied for finding accurate edge map to segment the image that produces variable number of regions.

**Keywords:** Segmentation, Grey Scale Average (GSA), Type-2 fuzzy logic, Fuzzy weighted average (FWA).

## 1 Introduction

Edge of an image represents the sudden change in the pixel intensity value creating two separate regions of different intensities. However, other factors like poor focus or refraction may be the cause of formation of an edge in the image [1]. Relatively early, the problem of image segmentation has been formalized by Health [5] as the minimization of an energy function that penalizes deviations from smoothness within regions and the length of their boundaries [4]. In the level set approach [2], $N$ regions are represented using exactly $\log_2 N$ level set functions. Grey scale average (GSA) method has been proposed here for finding accurate edge map to segment the image that produces variable number of regions [3], described in figure-1. The paper has been divided into five sections. Section 2 describes type-2 fuzzy logic to classify the color regions while in section 3, fuzzificaion and defuzzification method in image segmentation and how to determine the grey scale average has been discussed. The proposed algorithm with masking and GSA method is presented in section 4 followed by experimental results and discussions in section 5.

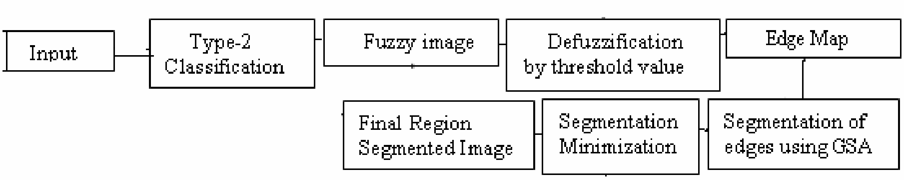


**Fig. 1.** Complete Block Diagram of the System

V.V. Das, G. Thomas, and F. Lumban Gaol (Eds.): AIM 2011, CCIS 147, pp. 277–280, 2011.

## 2 Classification of Image

Fuzzy Gaussian filter [6] is used to remove noise and sharpen the image, a preprocessing step of edge detection with the objective of enhancing the edges [7]. After removal of noise, the image is classified using type-2 fuzzy rule based classifier.

Let the pixel intensity of an input image $f(x,y)$ is represented by a vector $\boldsymbol{X} = (x_1, x_2, \ldots..x_m)$ where $m$ is the number of pixels in the image. Assume $r$ is the total number of fuzzy rules and one of such rule is represented as: *If* $x_1$ is $R_{j1}$ and $x_2$ is $R_{j2}$ and …..and $x_m$ is $R_{jm}$ *then* $X = (x_1, x_2, x_3, \ldots x_m)$ belongs to class $k$ with $CF = CF_j$ where $j=1,2,\ldots,r$ and $k=1,2,\ldots,M$ is the number of color classes classified using equation (1).

$$c = \arg\max_{j} G_i(x).CF_j \tag{1}$$

$CF_j \in [0,1]$ is the certainty factor of the $j$-th rule such that $G_i(x) = \prod_{i=1}^{m} \mu_R(x_i)$ where $\mu_R(x_i)$ is the degree of membership value of pixel $i$ in type-2 fuzzy set $R$.

## 3 Grayscale Weighted Average Method

The classified regions are now segmented by detecting edges using GSA method. All the edge points of the image constitute a set, called an edge map. Edge map is the specific region bounded by neighborhood pixels within the same object, shown in figure-2 where the color lines are the edges separated by different b/w color intensity.

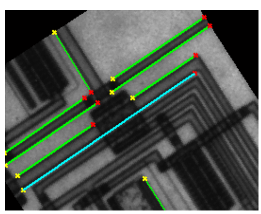

**Fig. 2.** Edge Map in B/W Image

Pixels with membership value '1'/'0' definitely belong/don't belong to the edge map set. However, pixels with intermediate membership values may or may not belong to the edge map, which are determined with certainty depending upon a prescribed threshold value. After thresholding, a binary image is obtained, which is the edge map representation of the original image.

The generalized version of the segmentation problem with an arbitrary number of regions $N$ is considered by providing the energy function of the model in (2).

$$E(\Omega_i, p_i, N) = \sum_{i=1}^{N} \int_{\Omega_i} \log p_i dx + \frac{\nu}{2} \int_{\Gamma_i} ds + \lambda \tag{2}$$

Where $\Omega_i$ = $i$-th region, $p_i$ = a priori probability of $i^{th}$ pixel, $\Gamma_i = i^{th}$ region boundary, $v$ = weighted parameter of boundary $\Gamma$, $ds$ is the deviation of distance between two region and the additional term of this energy functional penalizes the number of regions with the parameter λ. Starting with the entire image domain Ω as a single region, the two-region segmentation is applied in order to determine the best splitting of the domain. If energy decreases by splitting, two regions are formed, which are again divided and so on, until the energy does not decrease by further splits and thus the optimum number of regions is determined.

GSA deals with the grey scale value having the same range between 0-1 like FWA method. For the grey scale, the weighted average $GS_{avg}$ is computed as given in (3)

$$GS_{avg} = \sum_{i=1}^{255} w_i p_i \,/\, \sum_{i=1}^{255} w_i = f(w_1,...w_{255}, p_1,...p_{255}) \quad (3)$$

Where $w_i$ is the weighted intensity and $p_i$ is the pixel intensity.

## 4 Proposed Algorithm with GSA

```
Begin
Read input image f(x, y) of size M×N
Create the mask W (m×n) with mask coefficients, using
Sparse matrix so that sum of all coefficients of each
   Let mask=0 ;
     Mask weighted average, a=(m–1)/2   // small size
     Mask weighted average, b=(n–1)/2 // large size.
     g(x, y) = 0 // output image
   For y = b to (N – b – 1) do
      For x = a to (M – a – 1) do
         Calculate the (largest value) among all the maximum column values;
         Calculate the (smallest value) among all the minimum column values;
     GSA_range = (largest value) – (smallest value) / n
            For x = 1 to N do
                     for y = 1 to M do
                     g(x, y) = (f(x, y) – smallest pixel value) × 255 / GSA_range;
            End for
     Select a color image and convert it into grey scale input image;
     Store pixel values of the image along x and y coordinates in matrix form;
     Generate the Convolution mask for different gradient operators and store it ;
          SUM_coeffmatrix = 0; //Set all the points as black //
            Each mask along the horizontal and vertical direction is convolved
            with the input image; Calculate magnitude of the gradient vector;
          If E_thresold < 0.55 // Threshold is required to determine whether the point
          belongs to a specific region or not //
            Truncate unwanted edges from edge map information;
            Else include edge in the edge map set;
   End_for
End.
```

## 5 Conclusion and Result Discussion

The proposed algorithm is reasonably fast where 169 × 250 size image took 22.5 seconds on an Intel Atom 1.6 Ghz Processor having 1 GB of RAM size. By keeping the advantages of the level set framework, its main problem has been solved in the paper. The customized Gaussian filter [8] has a good contrast and sharpness characteristics, which is required to sharpen an image. This paper shows that the proposed segmentation model exhibits better performance in edge detection compare to the conventional method of segmentation [9].

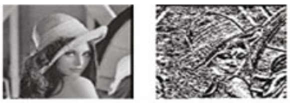

Fig – 2a Edge detection using Sobel Filter

Fig – 3 Edge detection by Custom Filter

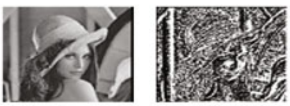

Fig-4 Segmentation by Conventional Method

Fig-5 Segmentation using Proposed Method

Fig-6 CMS

Fig-7 PMS

## References


[1] Argyle, E.: Techniques for edge detection. Proc. IEEE 59, 285–286 (1971)

[2] Chidiac, H., Ziou, D.: Classification of Image Edges. In: Vision Interface 1999, pp. 17–24. Troise-Rivieres, Canada (1999)

[3] Hueckel, M.: A local visual operator which recognizes edges and line. J. ACM 20(4), 634–647 (1973)

[4] Malik, J., Belongie, S., Shi, J., Leung, T.K.: Textons, contours and regions: cue integration in image segmentation. In: Proc. IEEE International Conference on Computer Vision, Corfu, Greece, pp. 918–925 (September 1999)

[5] Heath, M., Sarkar, S., Sanocki, T., Bowyer, K.W.: Comparison of Edge Detectors: A Methodology and Initial Study. Computer Vision and Image Understanding 69(1), 38–54 (1998)

[6] Shin, M.C., Goldgof, D., Bowyer, K.W.: Comparison of Edge Detector Performance through Use in an Object Recognition Task. Computer Vision and Image Understanding 84(1), 160–178 (2001)

[7] Peli, T., Malah, D.: A Study of Edge Detection Algorithms. Computer Graphics and Image Processing 20, 1–21 (1982)

[8] Gonzalez, R.C., Woods, R.E.: Digital Image Processing, 2nd edn. Prentice-Hall, Inc., Upper Saddle River (2002)

[9] Pratt, W.K.: Digital Image Processing, 4th edn. John Wiley & Sons, Inc., Hoboken (2007)