International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)
Volume 4, Issue 4, April 2015

Advanced Object Removal Using Super pixel Segmentation

Neethu Venugopal M, Sobin Francis

***Abstract*—Superpixels are defined as grouping uniform pixels in an image. Here superpixel segmentation is performed using the Lazy Random Walk (LRW) Algorithm. The method begins with initializing the seed positions and runs the LRW Algorithm on the input image. Then the boundaries of the superpixels are obtained using the optimization algorithm. Restoring the lost parts of an image and reconstructing them based on the background information is known as Inpainting. Here we provide a user interface where the user can select the portion of the image that is to be removed and then reconstruct it. Inpainting found main applications in the restoration of old photographs, removal of unwanted objects, etc.**

***Index Terms*—Segmentation, Superpixel, Lazy Random Walk, Optimization, Inpainting**

INTRODUCTION

Image segmentation is the process of dividing a digital image into multiple segments or into a set of pixels commonly known as superpixels. Segmentation enhances the appearance of an image so that it becomes more meaningful and easier to analyze. By using image segmentation we can trace objects and boundaries in images [10], [7]. Clearly saying, in image segmentation it assigns a label to each pixel in the image so that pixels with the same label share certain characteristics. The outcome of image segmentation is a set of segments. Each pixel in a region are similar with respect to some properties, such as color, intensity, or texture.

Grouping of uniform pixels in an image is commonly termed as superpixels. It has wide range of applications in computer vision. The intent of superpixel is to conceal the input image into an appressed manner. The aim of superpixel is to over-segment the input image into small dense regions with comparable appearances.

An image can be considered as a two-dimensional function f(x,y), where x and y are the spatial co-ordinates and the magnitude of 'f' at any point (x,y) gives the intensity or the gray level of the image, with the subsequent effects

1. ***Brightness:*** It is the perception elicited by the luminance of a visual target

2. ***Contrast:*** It is defined as the difference between the darkest and brightest areas of the image. More precisely, it is the difference in luminance or color that makes an object distinct.

3. ***Resolution:*** It is the detail an image holds. It is expressed in dots-per-inch (dpi) or pixels-per-inch (ppi).

4. ***Pixels:*** Generally thought of as the smallest single component of a digital image. It is the most basic unit of an image displayed on a computer or television screen. Pixels are generally arranged in rows and columns. A combination of pixels of various brightness and color values forms an image. Also known as pel or picture elements.

Our method begins with initializing the seed positions. Then we apply the LRW algorithm on the input image to obtain the probability of each pixel. Then we perform the optimization algorithm to obtain the boundary of the segmented image. Then we perform image inpainting. Here the user can select the regions to be removed and then we perform the SAD (Sum of Absolute Difference) Algorithm to remove the unwanted portions and then replace those regions with similar pixels from the background.

II. RELATED WORK

A large number of papers on superpixel segmentation have been developed in the last decade. The current superpixel approaches can be mainly divided into 2 categories. First category includes the algorithms which do not consider the compactness constraints while generating the superpixel. It includes mean shift algorithm [3], graph based algorithm [8]etc. These algorithms generally produce the superpixels by over segmenting the input image. The second category includes the algorithms which considers the compactness constraints such as normalized cuts [9], lattice cut[4], turbopixels [14], graph cut approaches [12] etc.

The approach to superpixel segmentation was first presented by Ren and Malik [9], where he used the normalized cuts (NCuts) algorithm. It is a very powerful method for obtaining superpixels of regular size and shape. When the size of the image increases the computational cost of this method increases. So it is not often used.

Levinshtein presented an efficient TurboPixel superpixel algorithm [14] using the level set based geometric flow evolution from the uniformly placed seeds in the image. However, it exhibited relatively poor boundary adherence.

There are other important superpixel approaches that have been proposed to fulfill the need of increasing applications, such as the algorithms in [5], [11], and [15].

In addition to these an important work related with our paper is the Random Walk Algorithm [10] which has a wide range of applications in the field of image processing. The related work includes [13], [6].

PROPOSED SYSTEM

This paper is an extension for image superpixel segmentation [10]. Over segmentation of the input image into small dense form with uniform appearance is the main objective of superpixel. Here each superpixel is designated with a unique label, so it can be considered as pixel labeling. We

ISSN: 2278 – 909X

begin by placing the initialized seeds of the destined superpixel. Then the LRW algorithm is applied to seize the initial superpixels. To make the superpixels more compact and their boundaries more steady, we developed the energy optimization algorithm.

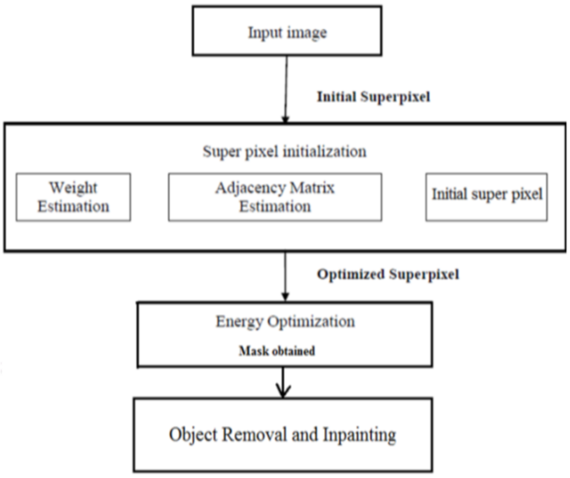


Fig 1.Block Diagram of proposed system

A. LAZY RANDOM WALK ALGORITHM

Interactive image segmentation is possible using the Random Walk Algorithm, which computes the first arrival probability[10], [13]. Here the Random Walk algorithm considers only the regional communication between the ongoing pixel and its corresponding seed. That is it bypasses the whole relationship between the current pixel and other seeds. So to have universal relationship between the pixel and all the seeds, we proposed the Lazy Random Walk (LRW) Algorithm, which computes the commute time from the seed points to other pixels.

Our method mainly consists of three steps:

1. Superpixel Initialization
2. Superpixel Optimization
3. Inpainting

SUPERPIXEL INITIALIZATION

This section describes the detection of the foreground and the background using the seeds specified by the user and then produces the initial superpixels.

The superpixel initialization is computed by the commute time. Commute time between two vertices u and v is defined as the expected time it takes the natural random walk starting in vertex u to travel to vertex v and then back to u. It is an Euclidean function, which considers all the path between u and v and not just the shortest path.

The main step in the initialization is the graph construction and the weight estimation. Here we construct a graph with nodes and edges. Each pixel in the image is represented by a node and the edges connect certain pair of neighboring pixels.

A. Graph Construction and Weight Estimation

We define our input image I(x,y) as a graph G=(V,E), which represents a weighted graph having a set of nodes V and edges E. The degree of each vertex is computed as:

$$d_i = \sum_j w_{ij} \tag{1}$$

Edge weight computation method is used to represent the changes in intensity of image pixels [9], [10], [6]. The closeness between two neighboring nodes is given by the edge weight. Thus w_{ij} is defined by the Gaussian weighting function given as:

$$W_{ij} = \exp\left[-\frac{\|g_i - g_j\|^2}{2\sigma^2}\right] \tag{2}$$

Where g_i and g_j gives the image intensity values at two nodes v_i and v_j and σ is the user defined parameter.

B. Adjacency Matrix

Then we define the adjacency matrix which gives the neighborhood relations [10]. Adjacency matrix is defined as:

$$W_{ij} = \begin{cases} 1-\alpha\,, if\ i=j \\ \alpha.w_{ij}\,, if\ i\sim j \\ 0\,, otherwise \end{cases} \tag{3}$$

Where i~j means nodes v_i and v_j are adjacent nodes and α is a control parameter in the range (0,1). Adjacency matrix is also known as the connectivity matrix. It gives the number of directed edges from v_i to v_j.

After that normalize the adjacency matrix to get the Transition Probability Matrix given by:

$$P_{ij} = \begin{cases} 1-\alpha\,, if\ i=j \\ \alpha.\frac{w_{ij}}{d_i}, if\ i\sim j \\ 0\,, otherwise \end{cases} \tag{4}$$

The above equation can also be written as:

$$P = (1-\alpha)I + \alpha D^{-1}W \tag{5}$$

Where D is a diagonal matrix and D_{ii} is the degree if the i^{th} vertex v_i. α is the sum of the weight of all the edges that is incident to v_i.

C. Laplacian Matrix

Next we define the laplacian matrix given by:

$$L_{ij} = \begin{cases} d_i\,, if\ i=j \\ -\alpha.w_{ij}\,, if\ i\sim j \\ 0\,, otherwise \end{cases} \tag{6}$$

It can also be written as: L = D – αW.

It assimilates a graph structure describing the local neighborhood relation between the data points. In a simple graph G with n vertices, its Laplacian matrix is defined as: L = D-A, where D is the degree matrix and A is the adjacency matrix of the graph. That is the neighborhood relation is given by the adjacency matrix.

We can define the normalized commute time from the Laplacian matrix as [16]:

$$CT_{ij} = \begin{cases} 1-L_{ij}^{-1}\,, if\ i\neq j \\ 1\,, if\ i=j \end{cases}$$

ISSN: 2278 – 909X

International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)
Volume 4, Issue 4, April 2015

The commute time is inversely proportional to the probability. We then construct the matrix S as:

$$S = D^{-1/2} W D^{-1/2} \qquad (7)$$

Finally we achieve the labels by computing $R(x_i) = \operatorname{argmax}(S_i)$ and then assigning label $R(x_i)$ to each pixel x_i. Then we obtain a $N\times 1$ coloumn vector as 0,0,…0,1,0,….,0 where all the elements are zero except the seed pixels. Then we obtain the superpixels by:

$S_{lk} = \{x_i / R(x_i) = l_k\}$ where (i=1,…N} and {k=1,…K}.

SUPERPIXEL OPTIMIZATION

Using the energy optimization function we improve the performance of the superpixels. It is improved by using the compactness constraints given as:

$$E = \sum_l (Area(S_l) - Area(S))^2 + \sum_l W_x CT(c_l^n, x)^2 \qquad (8)$$

The 1st term is the data item which optimizes the position of the seed points in order to make the superpixel boundaries adhere well to the object boundaries. The 2nd is the smooth item which divides the larger superpixels into smaller ones. . So here we calculate the compute time as C = 1-P. Then we calculate C_l using the equation:

$$C_l^n = \frac{\sum_l W_x \frac{CT(C_l^{n-1},x)}{\|x-C_l^{n-1}\|} x}{\sum_l W_x \frac{CT(C_l^{n-1},x)}{\|x-C_l^{n-1}\|}} \qquad (9)$$

In order to measure the texture information we use the texture feature of Local Binary Pattern (LBP). Here the output obtained will be a binary image. The area of the superpixel is obtained by the splitting and relocation mechanism. The 2 new superpixels obtained after splitting with two new centers is given by:

$$Cl_{new}, 1 = \frac{\sum_{\{(x-c_l).s>0\}} W_x \frac{CT(c_l,x)}{\|x-c_l\|} x}{\sum_{\{(x-c_l).s>0\}} W_x \frac{CT(c_l,x)}{\|x-c_l\|}} \qquad (10)$$

$$Cl_{new}, 2 = \frac{\sum_{\{x|x\in S_l,(x-c_l).s<0\}} W_x \frac{CT(c_l,x)}{\|x-c_l\|} x}{\sum_{\{x|x\in S_l,(x-c_l).s<0\}} W_x \frac{CT(c_l,x)}{\|x-c_l\|}} \qquad (11)$$

Calculate boundary each time in loop and when it satisfies the condition area of each superpixel will be optimized.

INPAINTING

Restoration of the lost part of an image and then reconstructing it based on the background information is known as inpainting. Here we are providing a user interface for image inpainting where the user can select the region to be removed and then reconstruct it with similar background pixels [18]. To perform image inpainting we are using the Sum of Absolute Difference (SAD) Algorithm. SAD Algorithm can be expressed as:

$$\sum_{(i,j)\in W} |I_1(i,j) - I_2(x+i, y+j)| \qquad (12)$$

SAD is the simplest algorithm used to measure the similarity between the pixels. It is calculated by subtracting pixels within a square neighborhood between the reference image I_1 and the target image I_2 followed by the aggregation of absolute difference, within the square window. Here if the left and the right images exactly match, the resultant will be zero. Here we are iteratively performing the SAD Algorithm to reconstruct the image with the background pixels.

Inpainting is mainly performed using the mask obtained in the optimization algorithm. Initially, the original image is padded with zeroes on both the sides. Then the image and the mask are divided into small blocks. Here we will be dividing into blocks of size 15 *15. After that the edge of the mask is detected using the canny edge detector. We will be dividing the mask along the edges. The R, G,B plane of each block is extracted. Afterwards each block of the mask is compared with the blocks in the original image. We use the SAD Algorithm to compare the blocks which then calculates the Euclidean distance. Then the block having the minimum Euclidean distance is taken from the original image blocks. Then each pixel in the mask is replaced with the pixels from the selected blocks of original image. This process is iteratively repeated in which the size of the mash or the object to be removed eventually decreases. Finally, the whole object will be removed and is replaced with similar pixels from the background.

EXPERIMENTAL RESULTS

Fig 1: Sample inputs

ISSN: 2278 – 909X

International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)
Volume 4, Issue 4, April 2015

Fig 2: Foreground and Background seed points

Fig 3: Initial Superpixels

Fig 4: Superpixels with centre points

Fig 5: Mask obtained

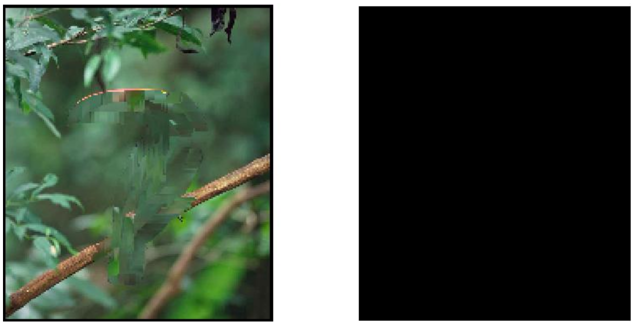
Fig 6: Optimized output

Fig 7: Region to be removed

(a)

(b)

ISSN: 2278 – 909X



(c)
Fig 8: Object Removal
(a): 1st iteration
(b): 2nd iteration
(c): 3rd iteration

Fig 9: Inpainted Image

CONCLUSION

In this paper we have presented an image superpixel segmentation based on the LRW and energy optimization algorithm and then inpainting is performed. We first perform the LRW algorithm to obtain the initial superpixels by placing the seed positions on the input image. Then we further improve the performance by the optimization algorithm which provides better boundary adherence, when compared with other known superpixel approaches. On the mask obtained we perform inpainting using the SAD Algorithm which helps in improving the picture quality by removing the unwanted object in the image.

ACKNOWLEDGMENT

First and foremost, praises and thanks to the God, the Almighty, for His showers of blessings throughout my project work to complete it successfully. First of all I would like to acknowledge with much appreciation the Pivotal role of the Principal of this institution, Prof. Shine P. James, who gave his full support to complete my work. At this moment of accomplishment, I pay homage to my guide, Asst.Prof.Sobin Francis (Electronics and Communication Engineering). This work would not have been possible without his guidance, support and encouragement. Under her guidance I successfully overcame many difficulties and learned a lot.

REFERENCES

[1] Lazy random walks for superpixel segmentation, Jianbing Shen, *Senior Member, IEEE*, Yunfan Du, Wenguan Wang, and Xuelong Li, *Fellow, IEEE*

[2] G. Mori, X. Ren, A. A. Efros, and J. Malik, “Recovering human body configurations: Combining segmentation and recognition,” in *Proc. IEEE*

[3] D. Comaniciu and P. Meer, “Mean shift: A robust approach toward feature space analysis,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 5, pp. 603–619, May 2002.

[4] A. Moore, S. Prince, J. Warrell, U. Mohammed, and G. Jones, “Superpixel lattices,” in *Proc. IEEE CVPR*, Jun. 2008, pp. 1–8.E.

[5] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fsua, and S. Süsstrunk, “SLIC superpixels,” EPFL, Lausanne, Switzerland, Tech. Rep. 149300, 2010.

[6] V. Gopalakrishnan, Y. Hu, and D. Rajan, “Random walks on graphs for salient object detection in images,” *IEEE Trans. Image Process.*, vol. 19, no. 12, pp. 3232–3242, Dec. 2010.

[7] Z. Li, X. M. Wu, and S. F. Chang, “Segmentation using superpixels: A bipartite graph partitioning approach,” in *Proc. IEEE CVPR*, Jun.2012, pp.789-796

[8] P. Felzenszwalb and D. Huttenlocher, “Efficient graph-based image segmentation,” *Int. J. Comput. Vis.*, vol. 59, no. 2, pp. 167–181, 2004.

[9] X. Ren and J. Malik, “Learning a classification model for segmentation,” in *Proc. 9th IEEE ICCV*, Oct. 2003, pp. 10–17.

[10] L. Grady, “Random walks for image segmentation,” *IEEE Trans. Pattern Anal. Mach Intell.*, vol. 28, no. 11, pp. 1768–1783, Nov. 2006.

[11] S. Xiang, C. Pan, F. Nie, and C. Zhang, “TurboPixel segmentation using eigen-images,” *IEEE Trans. Image Process.*, vol. 19, no. 11, pp. 3024–3034, Nov. 2010.

[12] O. Veksler, Y. Boykov, and P. Mehrani, “Superpixels and supervoxels in an energy optimization framework,” in *Proc. ECCV*, 2010, pp. 211–224.

[13] W. Yang, J. Cai, J. Zheng, and J. Luo, “User-friendly interactive image segmentation through unified combinatorial user inputs,” *IEEE Trans. Image Process.*, vol. 19, no. 9, pp. 2470–2479, Sep. 2010.

[14] A. Levinshtein, A. Stere, K. Kutulakos, D. Fleet, S. Dickinson, and K. Siddiqi, “Turbopixels: Fast superpixels using geometric flows,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 12, pp. 2290–2297,Dec. 2009

[15] Y. Zhang, R. I. Hartley, J. Mashford, and S. Burn, “Superpixels via pseudo-Boolean optimization,” in *Proc. IEEE ICCV*, Nov. 2011, pp. 1387–1394.

[16] J. Ham, D. D. Lee, S. Mika, and B. Sch¨olkopf, “A kernel view of the dimensionality reduction of manifolds,” in *Proc. 21st ICML*, 2004, pp. 1–9.

[17] R.C. Gonzalez and R.E. Woods, Digital Image Processing, 2nd ed. Pearson Education, 2002.

[18] Wen-Huang Cheng, Chun-Wei Hsieh, Sheng-Kai Lin, Chia-Wei Wang and Ja-Ling Wu. “Robust Algorithm for Exemplar-Based Image Inpainting”, in *The International Conference on Computer Graphics, Imaging and Vision (CGIV 2005), 26-29 July 2005, Beijing, China, pp. 64-69*

Neethu Venugopal received the degree in Electronics and Communication Engineering from University Of Kerala , Kerala , India.

Sobin Francis received the graduation in Electronics and Communication Engineering from University Of Kerala, kerala, india and post graduation in Microwave and Radar from Cochin University Of Science And Technology (CUSAT) , Kerala , India. He now works as an assistant professor at CUSAT.

ISSN: 2278 – 909X