Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 31 No. 5s (2024)

Optimizing Weight Functions for Enhanced Image Segmentation Using Normalized Cut

Subhashree Abinash[1,2], Sabyasachi Pattnaik[2]
[1] Synergy Institute of Engineering and Technology, Dhenkanal, India.
[2] Fakir Mohan University, Balasore, India,
https:/fmuniversity.nic.in

Article History:

Received: *10-05-2024*

Revised: *22-06-2024*

Accepted: *04-07-2024*

Abstract

Effective image segmentation remains a fundamental challenge in computer vision, with the Normalized Cut (Ncut) method emerging as a powerful technique for partitioning images into meaningful segments. The efficacy of Ncut largely depends on the choice of the weight function, which quantifies the similarity between image elements, be the pixels or predefined regions. This paper presents a novel framework to optimize the weight functions for Ncut in the context of image segmentation, aiming to bridge the gap between theoretical robustness and practical applicability. We first discussed the theoretical aspects of Ncut, emphasizing the role of weight functions in achieving segmentation that is both globally and locally suitable. Subsequently, we analyze the frameworks for the systematic selection of weight functions, effective to different image characteristics such as texture, color, and spatial relationships. Our methodology combining color spaces analysis, texture descriptors, and edge information. Through several experimentations on Corel and Berkley image segmentation datasets, including natural scenes and images, we demonstrate the comparisons of the weight functions over conventional methods in terms of segmentation quality and evaluated with standard algorithms like Otsu thresholding and C-means clustering algorithm. Three validity indices have been used to quantify the results and observe the superiority of the proposed model. This work not only advances the understanding of weight function optimization in Ncut-based image segmentation but also offers a practical guide for researchers and practitioners in computer vision.

Keywords: Image Segmentation, Normalized Cut, Weight Function, Computer Vision, Texture Analysis, Color Space, Edge Information.

1. Introduction

The Normalized Cut (Ncut) algorithm has been widely recognized for its effectiveness in image segmentation, leveraging graph theory to partition images into meaningful segments based on global image features. The algorithm's success in various applications underscores its versatility and efficiency in handling complex segmentation tasks [5]. However, the choice of weight functions within the Ncut framework significantly impacts its performance, necessitating a deeper exploration of optimization strategies to enhance segmentation outcomes. Recent advancements in weight function design have focused on incorporating multi-dimensional features such as color, texture, and spatial information to improve segmentation accuracy. These efforts underscore the importance of a comprehensive approach that considers the diverse characteristics of image data [6]. Moreover, the application of machine learning techniques for the automatic tuning of weight function parameters has shown promising results, offering a pathway to more adaptive and robust segmentation solutions [3]. In the domain of texture-based segmentation, the utilization of texture descriptors within weight functions has been particularly effective for distinguishing between complex patterns and surfaces [2,4]. This approach has enabled more nuanced differentiation between image regions, highlighting the critical role of texture analysis in segmentation tasks [4,16-17]. Furthermore, the integration of spatial proximity measures into weight functions has been shown to enhance the coherence of segmented regions, ensuring that segments are not only homogeneous in feature space but also contiguous in the spatial domain. This spatial consideration is vital for maintaining the integrity of object boundaries and facilitating the segmentation of closely situated but distinct entities [1,3,15]. The exploration of hybrid weight functions that combine multiple features and adapt to specific segmentation goals represents a significant area of ongoing research. These hybrid approaches aim to leverage the strengths of various descriptors and similarity measures, offering a more flexible and effective framework for addressing the challenges of image segmentation [8,12,18]. In conclusion, the literature underscores the critical importance of optimizing weight functions within the Ncut algorithm to achieve enhanced image segmentation. Future research directions include the further development of adaptive weight functions, the exploration of new feature descriptors, and the application of advanced machine learning techniques for parameter optimization.

The article is further divided into 4 sections, where the Section 2 explains the basic principle behind Graph Cut. Section 3 explains different existing weight functions and selection of the optimal weight for optimum result. Section 4 explains the results and discussions followed by conclusion in Section 5.

2. Graph Cut

Normalized Cut is an image partitioning approach proposed by Shi and Malik [5]. It is based on graph cut. This approach partitions a collection of pixels by representing them as nodes in a graph that is both undirected and weighted. In this context, the symbol 'V' represents a pixel or node, whereas 'E' represents the edge or the distance between two pixels. Next G=(V, E), an undirected weighted graph, is constructed, where a non-negative weight W is assigned to each edge.

The non-negative weight mentioned above is defined as [12]:

$$W(i,j) = e^{\frac{-\|F(i)-F(j)\|^2}{\sigma_I}} \times \begin{cases} e^{\frac{-\|X(i)-X(j)\|^2}{\sigma_X}} & if \quad \|X(i)-X(j)\|<r \\ 0 & otherwise \end{cases} \quad . \tag{1}$$

In (1) scenario, F(i) represents the feature derived from the intensity, while X(i) represents the spatial location of the ith node. The circle's radius is represented by the variable r, and the pixels are positioned inside this circle. If S1 and S2 are two sets of picture pixels that have no elements in common, then

$$S_1 \cup S_2 = V$$
$$S1 \cap S2 = \phi$$

The step-by-step procedure to cut the graph in graph cut is explained in Figure 1,

$$cut(S_1, S_2) = \sum_{i \in S_1, j \in S_2} W(i, j) \quad (2)$$

To partition the graph with optimum accuracy, the cut value mentioned in (2) should be minimize. Based on this concept, the graph cut has been redefined by Shi and Malik [5] and is given in (3).

$$Ncut(S_1, S_2) = \frac{cut(S_1, S_2)}{assoc(S_1, V)} + \frac{cut(S_2, S_1)}{assoc(S_2, V)} \quad (3)$$

In a graph, $assoc(S_1, V) = \sum_{i \in S_1, j \in V} W(i, j)$is the total number of connections from nodes in S1 to all the available nodes.

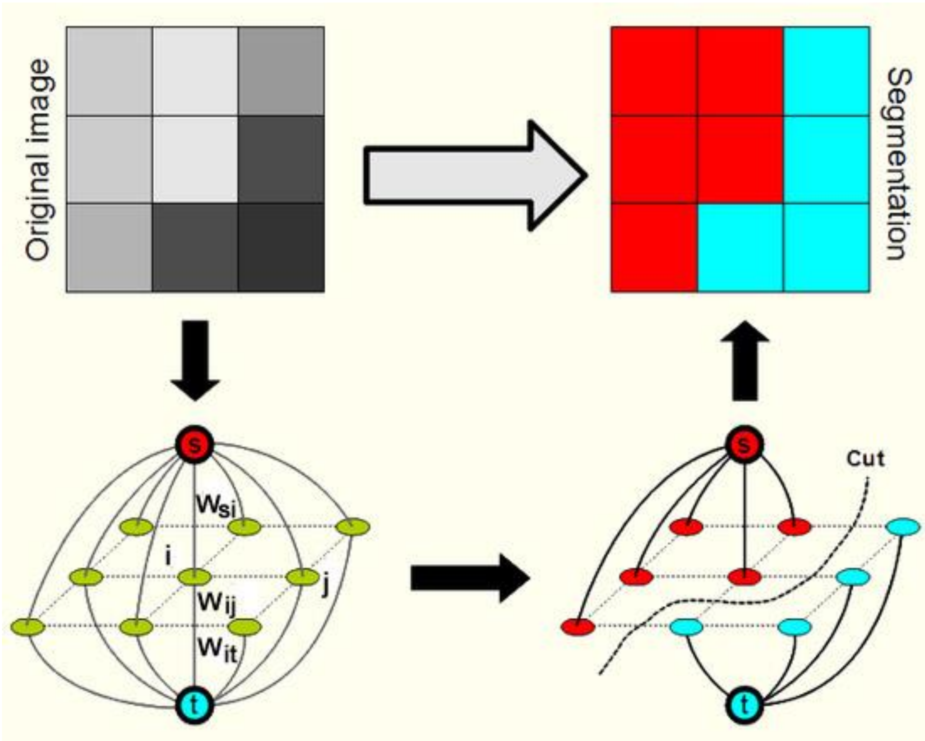

Fig. 1. Graph Cut

3. Proposed method for Selection of the weight functions

Choosing the right weight function for the Normalized Cut (Ncut) method in image segmentation is a critical step that directly influences the quality and relevance of the segmentation results. The weight function defines the similarity between pairs of pixels or regions, guiding the segmentation process to divide the image into coherent segments based on shared attributes such as color, texture, brightness, and spatial proximity. The selection and tuning of this function require a thoughtful consideration of the image characteristics and the specific goals of the segmentation task. The first step in selecting the right weight function is a thorough analysis of the image characteristics. Images can vastly differ in their content and structure, requiring different approaches for effective segmentation. Color is a primary feature in image segmentation. The choice of color space (RGB, HSV, Lab, etc.) and how color similarity is measured can significantly impact segmentation performance [1-4]. Texture provides critical information in images where color alone is not sufficient for differentiation. Texture analysis methods, such as Local Binary Patterns (LBP) or Gabor filters, can enhance segmentation by integrating texture-based similarity into the weight function [4]. The spatial proximity of pixels or regions plays a crucial role in segmenting images with coherent regions. Weight functions that incorporate spatial information can prevent the fragmentation of segments and maintain the continuity of structures [9,15]. Given the diversity of image characteristics, the selection of weight functions is not a one-size-fits-all solution. The choice depends on the specific requirements of the segmentation task and the characteristics of the image to be segmented.

3.1. Gaussian Similarity Function (GSF)

One of the most popular weight functions for image segmentation is the Gaussian similarity function. It is particularly effective for capturing the similarity based on intensity or color and spatial proximity (4):

$$W(i,j) = e^{\frac{-\|F(i)-F(j)\|^2}{\sigma_I}} \times \begin{cases} e^{\frac{-\|X(i)-X(j)\|^2}{\sigma_X}} & \text{if } \|X(i)-X(j)\|<r \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

F(i) and F(j) represent the intensity or color vectors of pixels i and j, respectively. X(i) and X(j) denote the spatial coordinates of pixels i and j, respectively. σ_I controls the sensitivity to differences in color or intensity, while σ_x adjusts the influence of spatial distance.

3.2. Gradient-Based Weight Function (GWF)

$$\boldsymbol{W(u,v) = e^{\frac{-\|\nabla I(u)-\nabla I(v)\|^2}{2\sigma^2{}_\nabla}} \times \begin{cases} e^{\frac{-\|X(u)-X(v)\|^2}{2\sigma^2{}_X}} & \text{if } \|X(u)-X(v)\|<r \\ 0 & \text{otherwise} \end{cases}} \tag{5}$$

This function (5) considers the gradient or the edge information between pixels, which is useful for detecting boundaries between segments. ∇I(u) and ∇I(v) are the gradient magnitudes at pixels u and v, indicating edge strength. σ_∇ controls the sensitivity to differences in gradient magnitude.

3.3. Texture Similarity Function(TSF)

$$\mathbf{w(u,v) = e^{\frac{-||T(u)-T(v)||^2}{2\sigma_T^2}}} \tag{6}$$

For the image where texture plays a significant role in differentiating regions, a texture similarity function might be employed, often based on texture descriptor such as local binary pattern (LBP) or Gabor filters. T(u) and T(v) represents the texture descriptors of pixels u and v, respectively. $\boldsymbol{\sigma_T}$ adjusts the sensitivity to difference in texture (6).

3.4. Combined Feature Similarity(CFS)

Combination of features (color, texture, gradient) is used to define the weight function, providing a more comprehensive measure of similarity (7).

$$\boldsymbol{W(u,v) = e^{\frac{-\|F(u)-F(v)\|^2}{2\sigma^2{}_F}} \times \begin{cases} e^{\frac{-\|X(u)-X(v)\|^2}{2\sigma^2{}_X}} & \text{if } \|X(u)-X(v)\|<r \\ 0 & \text{otherwise} \end{cases}} \tag{7}$$

F(u) and F(v) could include combined features such as color intensity, texture, and gradient.

σ_F and σ_X parameters that respectively control the sensitivity to feature differences and spatial proximity.

4. Results and Discussion

We did several tests with the four weight functions using the Normalized cut algorithm on the image's colour channels and we have selected the images from Corel and Berkley database. We have compared the results with Otsu and C Means algorithm and validated these results using Jaccard Coefficient, Dice coefficient and Mean Square error.

Let S1 be the segmented image and S2 be the ground truth. Both S1 and S2 have the same size, which is M x N. Then, JC is defined as (8):

$$J\,C(S_1, S_2) = \frac{S_1 \cap S_2}{S_1 \cup S_2} \tag{8}$$

DC is defined as (9):

$$Dice\,Coefficient(S_1, S_2) = \frac{2 \times |S_1 \cap S_2|}{|S_1| + |S_2|} \tag{9}$$

DC and JC values closure to unity proves the high efficiency of the model.

The formula for MSE (10),

$$MSE = \frac{1}{M \times N} \sum_{i=1}^{M \times N} |S_1(i) - S_2(i)|^2. \tag{10}$$

MSE values close to zero indicate high efficiency, while values close to unity indicate low efficiency.

Table-1 Jaccard Coefficient

Image SL No	GSF	GWF	TSF	CSF	Otsu	C -means
Berkley Image Dataset						
54011	0.89	0.92	0.93	0.92	0.51	0.27
16555	0.91	0.9	0.91	0.90	0.52	0.54
32651	0.89	0.9	0.93	0.91	0.83	0.84
36478	0.86	0.91	0.95	0.93	0.62	0.62
45121	0.89	0.88	0.91	0.92	0.49	0.47
14563	0.87	0.89	0.97	0.96	0.74	0.65
Corel Image Dataset						
11201	0.84	0.88	0.96	0.97	0.89	0.81
11224	0.86	0.89	0.94	0.95	0.43	0.46
11376	0.87	0.84	0.93	0.92	0.9	0.92
12456	0.85	0.86	0.94	0.95	0.34	0.63
12785	0.91	0.89	0.94	0.95	0.55	0.37
12459	0.89	0.87	0.95	0.96	0.34	0.33

Table-2 Dice coefficient

Image SL No	GSF	GWF	TSF	CSF	Otsu	C -means
Berkley Image Dataset						
54011	0.86	0.87	0.82	0.91	0.72	0.19
16555	0.85	0.86	0.87	0.91	0.41	0.37
32651	0.87	0.91	0.85	0.89	0.31	0.75
36478	0.84	0.89	0.83	0.9	0.33	0.58
45121	0.84	0.87	0.87	0.91	0.72	0.34
14563	0.89	0.88	0.91	0.9	0.41	0.54
Corel Image Dataset						
11201	0.81	0.87	0.84	0.9	0.79	0.73
11224	0.89	0.91	0.89	0.87	0.31	0.35
11376	0.93	0.93	0.87	0.88	0.81	0.87
12456	0.92	0.93	0.88	0.91	0.21	0.47
12785	0.88	0.93	0.83	0.89	0.39	0.21
12459	0.94	0.91	0.89	0.87	0.18	0.22

Table-3 Mean Square Error (MSE)

Image SL No	GSF	GWF	TSF	CSF	Otsu	C -means
Berkley Image Dataset						
54011	0.06	0.08	0.05	0.03	0.25	0.44
16555	0.05	0.09	0.03	0.04	0.16	0.15
32651	0.04	0.03	0.03	0.04	0.05	0.12
36478	0.03	0.04	0.04	0.05	0.06	0.35
45121	0.07	0.08	0.08	0.09	0.19	0.21
14563	0.06	0.03	0.05	0.04	0.07	0.12
Corel Image Dataset						
11201	0.07	0.08	0.08	0.09	0.19	0.21
11224	0.04	0.02	0.05	0.03	0.08	0.08
11376	0.03	0.04	0.02	0.03	0.30	0.05
12456	0.05	0.07	0.04	0.05	0.12	0.11
12785	0.06	0.07	0.07	0.09	0.27	0.48
12459	0.03	0.04	0.02	0.03	0.58	0.56

Table 1 presents the computed values for the Jaccard Coefficient of the segmented pictures. The values of Otsu and C Means in Table 1 are close to zero due to the algorithms difficulty to accurately identify the region of interest in the given images. This indicates that these techniques are not very efficient for the selected images. However, the values of the weight functions on Normalised cut algorithm are nearly equal to one, indicating that the weight functions on Normalised cut are more effective than the standard methods for all the chosen images. Table 2 and Table 3 provide the Dice

Coefficient and Mean Square Error (MSE) values for the chosen photos from the Berkley and Corel collection. The plot of the MSE mentioned in Table 3 is given in Figure 4. The values in the tables demonstrate that the four-weight function-based models outperform the standard techniques, Otsu, and C Means, for the selected images from the Berkley and Corel Image segmentation dataset.

a b c d

Fig.2. Berkley image segmentation dataset (a, b) original image and ground truth, (c, d) Otsu and C-means.

a b c d

Fig.3. Original picture, ground truth, Otsu, C-means (Corel image segmentation dataset)

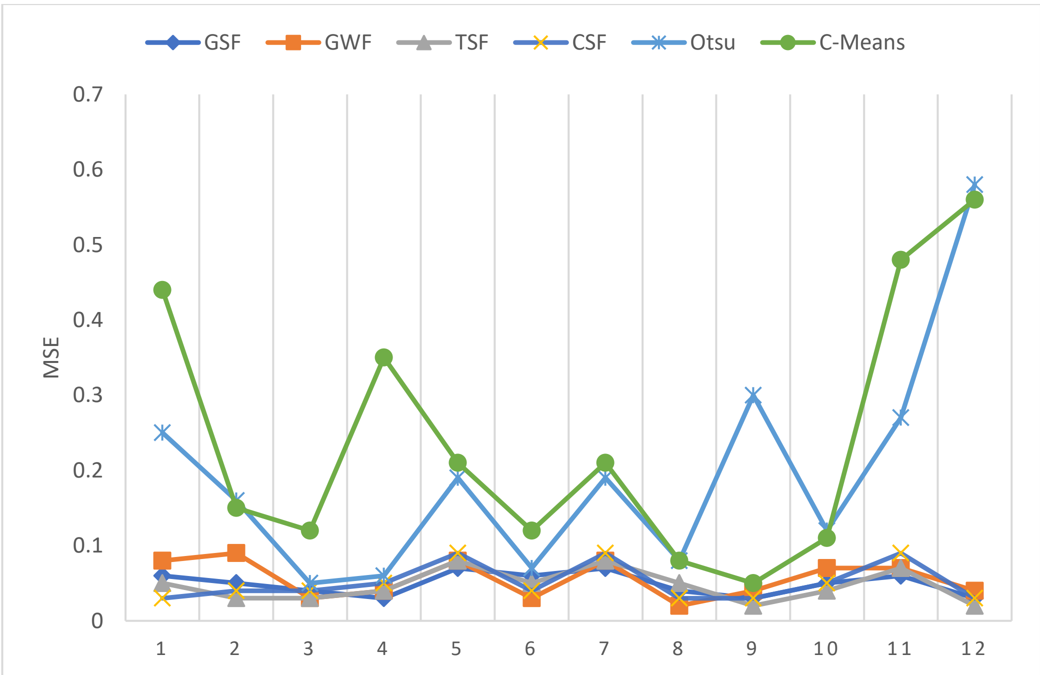

Fig.4. MSE plot for different models

The main goal of this work was to combine the different methods and improve weight functions in the context of the Normalized Cut (Ncut) technique for picture segmentation. The investigation encompassed four weight functions: Gaussian, gradient-based, texture similarity function, and combined feature similarity functions. Each weight function was designed to improve the quality of segmentation by adjusting to specific image characteristics, including texture, color, and spatial relationships. A key finding of this work is that gradient-based weight functions improve Ncut edge identification. By directly adding gradient information into weight calculation, these methods improve picture boundary and texture detection. Gradient-based functions perform best in images with well-defined edges, but they may struggle in images with less defined edges or when texture and color dominate.

5. Conclusion

This work has presented a thorough examination of the techniques and progress made in enhancing weight functions for the Normalized Cut (Ncut) method, a crucial approach in the domain of image segmentation. The investigation demonstrates that the efficacy of the Ncut technique in picture segmentation is significantly impacted by the selection and optimization of weight functions. Conventional weight functions that are designed to work for all images are being replaced by advanced, adaptive techniques. These strategies can dynamically adapt to the specific characteristics of each image, resulting in improved accuracy and efficiency in image segmentation. The optimization of weight functions for the Normalized Cut method is a dynamic and expanding area of research that is essential for the advancement of picture segmentation technology. The current progress sets the stage for future advancements, which have the potential to unlock new possibilities in a range of fields, including medical diagnostics and autonomous car navigation.

References

[1] Comaniciu, D., & Meer, P. (2002). Mean shift: A robust approach toward feature space analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(5), 603-619.

[2] Haralick, R. M., Shanmugam, K., & Dinstein, I. (1973). Textural features for image classification. IEEE Transactions on Systems, Man, and Cybernetics, 3(6), 610-621.

[3] Wang, F., Zhao, C., Liu, J., & Huang, H. (2020). A variational image segmentation model based on normalized cut with adaptive similarity and spatial regularization. SIAM Journal on Imaging Sciences, 13(2), 651-684.

[4] Ojala, T., Pietikäinen, M., & Mäenpää, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(7), 971-987.

[5] Shi, J., & Malik, J. (2000). Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(8), 888-905.

[6] Wang, J. Z., & Siskind, J. M. (2003). Image segmentation with ratio cut. IEEE Transactions on Pattern Analysis and Machine Intelligence, 25(6), 675-690.

[7] Muda, T. Z. T., & Salam, R. A. (2011, November). Blood cell image segmentation using hybrid k-means and median-cut algorithms. In 2011 IEEE International Conference on Control System, Computing and Engineering, pp. 237-243.

[8] Felzenszwalb, P. F., & Huttenlocher, D. P. (2004). Efficient graph-based image segmentation. International Journal of Computer Vision, 59(2), 167-181.

[9] Goldberg, D. E. (1994). Genetic and evolutionary algorithms come of age. Communications of the ACM, 37(3), 113-120.

[10] C. Yang, L. Zhang, H. Lu, X. Ruan and M. Yang, ”Saliency Detection via Graph-Based Manifold Ranking,” 2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, 2013, pp. 3166-3173.

[11] B. Jiang, L. Zhang, H. Lu, C. Yang and M. Yang, ”Saliency Detection via Absorbing Markov Chain,” 2013 IEEE International Conference on Computer Vision, Sydney, NSW, 2013, pp. 1665-1672.

[12] S. Abinash, “Two Novel Graph Theory Based Algorithm for Salient Object Detection.” 2018 IEEE International Conference on Computing Communication and Automation, New Delhi, 2018, pp.1-6.

[13] V. Navalpakkam and L. Itti, “An Integrated Model of Top-Down and Bottom-Up Attention for Optimizing Detection Speed,” Proc. IEEE CS Conf. Computer Vision and Pattern Recognition, pp. 2049- 2056, 2006.

[14] U. Rutishauser, D. Walther, C. Koch, and P. Perona, “Is BottomUp Attention Useful for Object Recognition?” Proc. IEEE CS Conf. Computer Vision and Pattern Recognition, pp. 37-44, 2004.

[15] D. Walther, L. Itti, M. Riesenhuber, T. Poggio, and C. Koch, “Attentional Selection for Object Recognition—A Gentle Way,” Proc. Second Int’l Workshop Biologically Motivated Computer Vision, 2002.

[16] Ke Yan, Changyang Li, Xiuying Wang, Ang Li, Yuchen Yuan, Jinman Kim, Dagan Feng, "Adaptive background search and foreground estimation for saliency detection via comprehensive autoencoder", Image Processing (ICIP) 2016 IEEE International Conference on, pp. 2767-2771, 2016.