REPUBLIC OF TÜRKİYE
ADANA ALPARSLAN TÜRKEŞ SCIENCE AND TECHNOLOGY UNIVERSITY

GRADUATE SCHOOL
ELECTRICAL AND ELECTRONIC ENGINEERING DEPARTMENT

IMAGE SEGMENTATION TECHNIQUES VIA ROBUST HYPOTHESIS TESTING

SHAHIN MAMMADOV

M.Sc.

ADANA 2023

REPUBLIC OF TÜRKİYE

ADANA ALPARSLAN TÜRKEŞ SCIENCE AND TECHNOLOGY UNIVERSITY

GRADUATE SCHOOL

ELECTRICAL AND ELECTRONIC ENGINEERING DEPARTMENT

IMAGE SEGMENTATION TECHNIQUES VIA ROBUST HYPOTHESIS TESTING

SHAHIN MAMMADOV

M.Sc.

THESIS ADVISOR

ASST. PROF. DR. HÜSEYİN AFŞER

ADANA, 2023

DECLARATION OF CONFORMITY

In this thesis study, which was prepared following the thesis writing rules of Adana Alparslan Türkeş Science and Technology University Institute of Graduate School, I declare that I provide all the information, documents, evaluations, and results in accordance with scientific ethics and moral codes without resorting to any means or assistance that would be contrary to scientific ethics and traditions. I also declare that I refer to all of the articles I used in this study with appropriate references and accept all moral and legal consequences if a situation is found contrary to my statement regarding my work.

09/01/2023

[Signature]

Shahin MAMMADOV

ABSTRACT

IMAGE SEGMENTATION TECHNIQUES VIA ROBUST HYPOTHESIS TESTING

Shahin MAMMADOV

M.Sc., Department of Electrical and Electronics Engineering

Supervisor: Asst. Prof. Dr. Hüseyin AFŞER

January 2023, 38 Pages

In this thesis, the multiple segmentation of color images is examined using the method based on robust hypothesis testing. The robust hypothesis test has been adapted to image segmentation by several different methods. In the first approach, to ease this NP-hard problem, we provide the user to choose certain areas over the image whose pixels are regarded as candidates from different labels. The suggested algorithm is started through this selection process, and the empirical histograms of the pixel densities are then employed as nominal densities in a reliable hypothesis test. By modifying the test framework to include soft metrics obtained over the multi-dimensional color image, we specifically apply the DGL test. In other ways, we included superpixels in our test. But just adding superpixels, the baseline method, does not take advantage of the spatial information of user input, and only works in the color domain. In addition to superpixels, we have applied the DGL test based on spatial-temporal distribution, where the spatial information of bounding boxes is included in the test for improved performance without increasing the complexity. We also provided simulations on Berkeley's BSDS500 image dataset and showed that the last method applied could result in a 10% improvement in segmentation accuracy to the first method. The methods we develop can multiple segmentation color images with low complexity. This performance advantage has also been verified by simulations.

Keywords: User-Assisted Segmentation, Color Image Segmentation, Multiple Segmentation, Robust Hypothesis Testing, DGL Test

ÖZET

KARARLI HİPOTEZ TESTİ İLE GÖRÜNTÜ BÖLÜTLEME TEKNİKLERİ

Shahin MAMMADOV
Yüksek Lisans, Elektrik ve Elektronik Mühendisliği Anabilim Dalı
Danışman: Dr. Öğr. Üyesi Hüseyin AFŞER
Ocak 2023, 38 Sayfa

Bu tezde, kararlı hipotez testine dayalı yöntem kullanılarak renkli görüntülerin çoklu kesimlenmesi incelenmiştir. Kararlı hipotez testi, birkaç farklı yöntemle görüntü bölütlendirmeye uyarlanmıştır. İlk yaklaşımda, NP-zorluğa sahip bu problemin çözümü için kullanıcı yardımıyla farklı etiketlerden gelen piksellerin kabaca seçildiğini farz ettik. Önerilen algoritma bu seçimle başlatılır ve piksel yoğunluklarının ampirik histogramları daha sonra kararlı bir hipotez testinde nominal yoğunluklar olarak kullanılır. Test çerçevesini çok boyutlu renkli görüntü üzerinden elde edilen hassas ölçümleri içerecek şekilde değiştirerek, DGL testini uyguluyoruz. Sonraki yöntemlerimize süperpikselleri dahil ettik. Ancak temel yöntem olan süperpikselleri eklemek, kullanıcı girişinin uzamsal bilgisinden yararlanmaz ve yalnızca renk alanında çalışır. Süperpiksellere ek olarak, karmaşıklığı artırmadan geliştirilmiş performans için sınırlayıcı kutuların uzamsal bilgilerinin teste dahil edildiği uzamsal-zamansal dağılıma dayalı DGL testini uyguladık. Ayrıca Berkeley'in BSDS500 görüntü veri seti üzerinde simülasyonlar sağladık ve uygulanan son yöntem, ilk yönteme göre segmentasyon doğruluğunda %10'luk bir iyileşmeyle sonuçlandı. Geliştirdiğimiz yöntemler, düşük karmaşıklığa sahip renkli görüntüleri çoklu segmentasyon haline getirebilir. Bu performans avantajı simülasyonlarla da doğrulanmıştır.

Anahtar Kelimeler: Kullanıcı Destekli Segmentasyon, Renkli Görüntü Kesimleme, Çoklu Kesimleme, Kararlı Hipotez Testi, DGL Testi

ACKNOWLEDGEMENTS

I thank my supervisor, Asst. Prof. Dr. Hüseyin AFŞER, for providing guidance and feedback throughout this thesis.

I would also like to thank my thesis committee members, Assoc. Prof. Zekeriya TÜFEKCİ, Asst. Prof. Dr. Gökay DİŞKEN, for their constructive comments.

Finally, I must express my very profound gratitude to all my beloved family for providing me with unfailing support and continuous encouragement through writing this thesis.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

DGL	:Devroye, Gyorfi and Lugasi
HSV	:Hue, Saturation, Value
RGB	:Red, Green, Blue
LAB	:Lightness, Red/Green Value, Blue/Yellow Value
FLDA	:Fishers Linear Discriminant Analysis
MRF	:Markov Random Fields
NC	:Normalized Cut
CNN	:Convolutional Neural Networks
PDE	:Partial Differential Equation
DNN	:Deep Neural Networks
RG	:Region Growing
GC	:Garph-Cut
RW	:Random Walk
ACP	:Adaptive Constraint Propagation
SDG	:Spatial Distribution of Gaussians
CRF	:Conditional Random Field
DRF	:Deep Random Field
GT	:Ground Truth
SLIC	:Simple Linear Iterative Clustering
BB	:Bounding Box

LIST OF SYMBOLS

H_j	:Hypothesis j
H_i	:Hypothesis i
P	:Probability distribution of P
Q	:Probability distribution of Q
$\mathscr{X}$	:Discrete alphabet of probability distributions
H	:Entropy of the probability distribution
Ω,M	:Mutually exclusive and collectively exhaustive acceptance region
A	:Borel set
$\mathscr{A}$	:Collection of $k(k-1)/2$ Borel sets
Ψ,$\mathbb{I}$	:Indicator function
N	:Number pf pixels
I	:Image
S	:Superpixels
K	:Number of superpixels
D,d	:Distance
m,w_j	:Weight terms
L,T	:Labeled pixels

1. INTRODUCTION

To develop and process the information from the given image without altering the other properties of that image is one of the most difficult phases of image processing [1, 2]. One of the key areas of image processing is image segmentation, which involves segmenting an image into several portions. Each component informs the user about things like color, intensity, or texture. As a result, it's crucial to use segments to isolate any image's bounds [3]. Three aspects of the image "color, intensity, and texture" are used to distinguish between various areas of the image [4].

Image segmentation is crucial in practically every area of science, including biometry, satellite imaging, machine vision, computer vision, and medical imaging, so its significance cannot be overstated [5, 6]. It is believed that there is no ideal, flawless approach for image segmentation because each image has a different kind [7]. Consequently, there are numerous segmentation methods.

In this thesis, we performed image segmentation using Bayes' multiple hypothesis test. The best-known robust analysis for multiple hypothesis testing is the DGL method. The DGL method analyzes multiple hypothesis testing problems defined for continuous and discrete cases. The DGL method is explained in Section 3.1. Our method has been improved each time to get more accurate results. Three different methods are presented in this thesis.

Firstly, our way of segmenting the image by sliding windows was done. In this method, the DGL test is applied by dividing the image into squares. The horizontal (h_l) and vertical (v_l) lengths are determined and the squares on the image move according to these lengths. At the same time, the sliding windows method was made in two separate parts soft sliding and hard sliding. The windows set in the hard method move only by sliding from side to side, while in the soft method, they overlap and slide side to side at the same time. This method is explained in detail in Section 3.2.

In the other method, we met superpixels and realized that if we add these superpixels to our method, we will get even better results and our resulting image will not appear square by square. And this method is combined with superpixels. After combining the method with the superpixels, even better results were obtained by including the center pixel of the superpixels. Superpixels are presented in Section 3.3 and the methods of adding them are presented in Section 3.4.

In this thesis, we added bounding boxes to our last method. Considering the joint probabilities of these boxes, we also included the spatial values. Now, the DGL test, which we applied in our thesis, calculates the spatial values as well as the color. Depending on both the color and the spatial, our images are better segmented and give better accuracies. This method is explained in Section 3.5.

When we want to distinguish an object of a certain color in an image segmentation application, it is more convenient to use the HSV color space. Because, unlike RGB, we can distinguish colors more clearly by applying a threshold value using only the tone value. Therefore, in this thesis, HSV color space is used, not RGB.

The above image segmentation methods, superpixels, and DGL method is theoretically explained in Chapter 3, and then results and simulations of image segmentations are introduced in Chapter 4. Finally, the results of this thesis are concluded in Chapter 5.

2. LITERATURE REVIEW

Using similar or distinctive features, segmentation divides an image into areas or objects [8]. Some examples of image segmentation methods include:

Threshold-Based Image Segmentation: Regions are divided based on histogram intensity. Threshold segmentation can be done using a variety of methodologies and approaches, including the Mean, Pile, and Otsu techniques [9].

Region-based segmentation is noise resilient, which is one of its best qualities. Based on predetermined criteria, such as color, intensity, or object, it divides an image into various zones. The three basic kinds of region-based segmentation techniques are region growing, region splitting, and region merging. [10].

Region Growing: Begins with a pixel and iteratively adds pixels to the region based on similarity until all of the pixels in the region are present [11–13]. When the image is noisy or contains intensity changes, it over segments and is unable to distinguish the shadow of the actual image [14].

Split-Merge: Initially captured as a single location, the complete image is periodically separated until no further splits are feasible. A division structure is a quadtree. The process is then repeated if the two regions are close to one another and similar until no further merger is possible [11–13]. The image's position and orientation cause a block final segmentation, while the regular partition causes over-segmentation [11].

Edge-Based Image Segmentation: The first step in image segmentation is edge detection. divides the foreground and background of an image. The splitting procedure is carried out using the edge detection approach based on the image's intensity or pixel change. uses two basic techniques. These are the gradient and gray histogram [15, 16]. Techniques based on gradients: Sobel, Prewitt, and Robert operators [17]. Techniques based on Gaussians: Canny edge detection and Laplacian of Gauss [17].

K-Means Clustering: On clustering, this segmentation technique is built. Additionally, it is a statistical technique for segmenting images. By adding p points to the cluster where the difference between the point and the mean is the smallest, divide an image into K clusters [11, 18]. Presumes clearly defined borders between clusters [12]. K-Mean Clustering is not effective in continuous regions [14].

Normalized Cuts: Reduces the number of areas in order to focus on the best regions. In a graph, each pixel serves as a vertex. Adjacent pixels are linked by edges. Edge weights between two matching pixels are determined by their resemblance, proximity, color, gray level, or textures. [11, 12, 19]. Its calculation is highly intricate. [11]. Lazy snapping and Graph-Cuts are methods of normalized cuts.

Lazy Snapping: The process of cutting off an object from an image is known as image cutout. At the single-pixel level, the foreground and background can be specified. The goal of Image Cutout is to distinguish between the foreground and background of an image. Normal cutting requires the user to specify pixel precision and each foreground area independently. However, choosing a region in Lazy Snapping will suffice. But Lazy Snapping is divided into two parts: Boundary editing and object marking [20]. In other words, the background and foreground are blended one after the other even after the items have been chosen, and it should be emphasized that the user has a lot of work to do in this situation as well.

Graph-Cuts: Background and foreground are linked to each pixel. These grounds should be divided for segmentation. Another image cutout method is Graph-Cuts. When the texture or color of both the foreground and background are fairly complex, threshold and simple algorithms will not work. When removed, a cut creates a line dividing the foreground from the backdrop. With this method, pixels are connected to one another and to terminals. Back foreground and background terminals are present. Find the minimum cut by giving each edge of both connections a weight [21, 22]. It is not ideal to scribble more.

ANN-Based Image Segmentation: Each neuron in an artificial neural network corresponds to a pixel in an image. The neural network is mapped to the image. Using training samples, a neural network is taught to create an image, and the neurons are then connected. The trained image is used to classify new photos [23].

CNN-Based Image Segmentation: Because of its great accuracy, convolutional neural networks are utilized for picture categorization and recognition. It is the process of assigning a distinct category to each pixel in the image. When it comes to the deep learning domain: semantic division [24].

Table 2.1: The summary of papers analyses.

	Papers	$M = 2$	$M > 2$	*ComplexityWithAnalyse*	*ComplexitywithTime*	*Robustness*	*Method*
1	[25]	✓		✓		✓	*Region Growing*
2	[26]		✓	✓			$Graph-Cutss$
3	[27]	✓					*PDE*
4	[28]	✓			✓	✓	*DNN*
5	[29]	✓			✓	✓	*CNN*
6	[30]	✓		✓	✓		*CNN*
7	[31]	✓		✓			$Graph-Cuts$
8	[32]		✓		✓		$Graph-Cuts$
9	[33]	✓				✓	$K-Means$
10	[34]	✓			✓	✓	$Graph-Cuts$
11	[35]	✓		✓	✓		$Graph-Cuts$
12	[36]	✓					*RG&GC*
13	[37]	✓				✓	*SDG*
14	[38]	✓		✓		✓	$Graph-Cuts\&RW$
15	[39]	✓				✓	$Graph-Cuts$
16	[40]	✓				✓	*Region Based*
17	[41]	✓		✓	✓		*CRF*

	Papers	$M = 2$	$M > 2$	ComplexityWithAnalyse	ComplexitywithTime	Robustness	Method
18	[42]		✓	✓	✓	✓	MRF
19	[43]	✓		✓	✓	✓	$ACP - Cut$
20	[44]	✓				✓	$Cascaded$
21	[45]	✓		✓	✓	✓	$FLDA$
22	[46]	✓		✓			$Laplacian$
23	[47]	✓				✓	$Random - Walks$
24	[48]		✓	✓			$Graph - Cuts$
25	[49]	✓		✓		✓	$MRF\&NC$
26	[50]	✓		✓	✓	✓	$Normalized - Cuts$
27	[51]	✓					$Graph - Cuts$
28	[52]	✓		✓	✓		$Graph - Cuts$
29	[53]	✓				✓	$Laplacian$
30	[54]	✓					$Graph - Cuts$
31	[55]	✓					$Guassian$
32	[56]	✓		✓			$Random - Walks$
33	[57]	✓					$Graph - Cuts$
34	[58]	✓		✓	✓	✓	$Graph - Cuts$

	Papers	$M=2$	$M>2$	*ComplexityWithAnalyse*	*ComplexitywithTime*	*Robustness*	*Method*
35	[59]	✓		✓	✓	✓	*Graph – Cuts*
36	[60]	✓		✓		✓	*Random – Walks*
37	[61]	✓					*Random – Walks*
38	[62]		✓				*Random – Walks*
39	[63]	✓		✓		✓	*Normalized – Cuts*
40	[64]		✓			✓	*Graph – Cuts*
41	[65]		✓	✓	✓		*DRF Model*
42	[66]	✓				✓	*Graph – Cuts*
43	[67]	✓				✓	*Graph – Cuts*
44	[68]	✓			✓	✓	*Graph – Cuts*
45	[69]	✓			✓	✓	*Region Merging*
46	[70]	✓			✓	✓	*Region Merging*
47	[71]	✓				✓	*Graph – Cuts*
48	[72]	✓			✓	✓	*Graph – Cuts*
49	[73]	✓				✓	*Graph – Cuts*
50	[74]	✓		✓	✓		*Graph – Cuts*
51	[75]	✓		✓		✓	*CNN*
52	[76]	✓		✓	✓		*Graph – Cuts*

3. MATERIALS AND METHODS

The hypothesis testing problem is one of the fundamental problems in decision theory. There are many applications connected to this problem like engineering, digital communications, image processing, and control. Hypothesis tests are designed based on a statistical model with the aim of minimizing error probability [77].

The necessity for robust hypothesis testing is caused by a statistical model that can derive significant losses on performance. In many applications where the consequences of an incorrect decision can be terrible, like earthquake detection, robust tests are needful. But robust tests have a weak point in which tests need to some performance sacrifices against the optimum test designed for a nominal model.

In the classical robust test, the minimax method is used. This method minimizes the error probability for the worst-case parameters, which is maximized the error probability [78].

3.1. Robust Bayesian Multiple Hypothesis Testing

In this section, the DGL method is explained for the defined problem in Section 3.1.1. Afterward, the different image segmentation methods were described in Sections 3.2, 3.3.1, 3.4.1, 3.4.2, 3.5 about sliding windows, superpixels, adding superpixels to the test, centroid of labeled pixels and superpixels, spatially varying segmentation respectively.

3.1.1. Method of DGL

In order to solve the difficulty of hypothesis testing, this method submits an exponential upper bound to a nonasymptotic uniform distribution [79]. The following are the basic explanations of the DGL approach.

$$A_{i,j} = \{x : Q_i(x) > x : Q_j(x)\}, 1 \leq i < j \leq k\}. \tag{3.1}$$

$\mathscr{A}$ denotes the collection of $k(k-1)/2$ sets of the form Equation 3.1 and A is a Borel set.

DGL developed the empirical measure to define the DGL test

$$\mu_n(A) = \frac{1}{n}\sum_{i=1}^{n} \Psi_{X_i \in \mathscr{A}} \tag{3.2}$$

where Ψ stands for the indicator function. This function compares distribution vector experimental observations made for each hypothesis using the set of A. In the next phase, the below decision rule is applied to this compared results. Then decision rule chooses a hypothesis as the final result.

$$\max_{A \in \mathscr{A}} |\int_A Q_j - \mu_n(A)| = \min_{i=1,..,k} \max_{A \in \mathscr{A}} |\int_A Q_i - \mu_n(A)| \tag{3.3}$$

For the DGL test, an exponential upper bound is [79],

$$P(e) \leq 2k(k-1)^2 e^{-n\varepsilon^2/2}. \tag{3.4}$$

3.2. Sliding Window DGL Based Segmentation

As many regions as the user wishes to segment the image into are chosen from the image in order to segment it into many sections. When sectioning, we use the technique of adding frames to the image and making sure that they appear immediately after the first window by window. The frame's dimensions are specified by the user. This segmentation is the Hard DGL approach. Additionally, we have made sure that the selected frame advances in steps rather than windows at a time. These images were created using this technique. The soft DGL approach is this. The user has the freedom to choose the steps and the frame. Our approach is therefore weakly supervised. The user's choices also have an impact on how accurately the image is scaled.

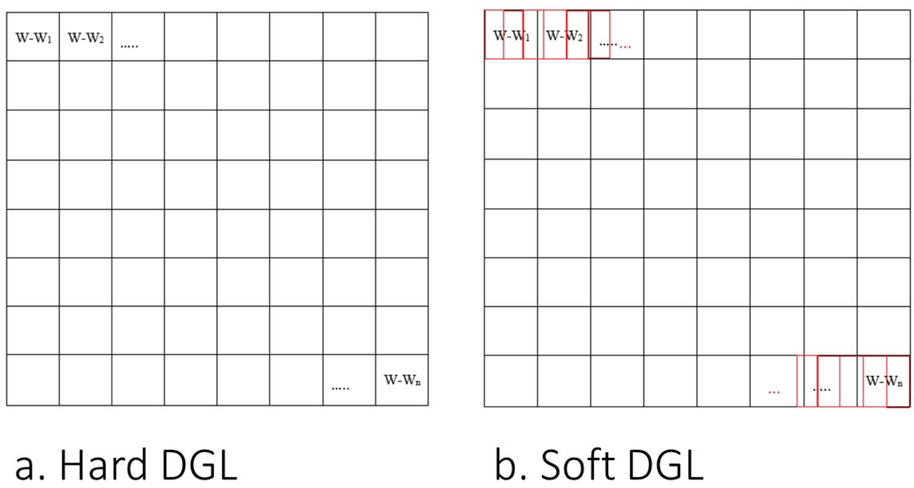

Figure 3.1: a.The Hard DGL approach is window-by-window. b.The Soft DGL technique moves from window to window while also moving step by step. Windows are the dark lines. Steps are the red lines. Both of them have user-defined sizes.

Think that we have $M*N$ image we want to segment it. $I = n$. It means image I is equal to n pixels. First, the user defines the window length horizontally (h_l) and vertically (v_l).

$$n = h_l * v_l. \tag{3.5}$$

The chosen region is subjected to the DGL test after the window sizes have been established, and a label is assigned based on the outcome. During the decision-making phase, DGL evaluates the distances between each hypothesis and selects the value with the smallest distance. It is Hard DGL. While the computed distances overlap, or add up, in the soft DGL test technique, several tests are applied by sliding windows at the same time, and our test selects the minimum sum of the overall distance values. With the soft DGL test, we hope to provide a little bit better performance and smoother operation than hard DGL. Briefly, we divided it into sliding windows through n pixels.

3.3. Superpixels

3.3.1. SLIC superpixels

Unlike other approaches, the SLIC superpixel is a better way to produce superpixels that increase the performance of faster, more efficient memory, cutting-edge boundary matching, and segmentation algorithms. *Simple linear iterative clustering* is a method of generating superpixels that differs significantly from k-means in two important ways:

1. The amount of distance calculations required for the optimization is significantly decreased by restricting the search area to a region proportional to the superpixel size. This complexity is reduced by the number of the linear pixels N and the superpixels number k [80].

2. Color and spatial closeness are combined with a weighted distance measure, which also allows for control over the size and compactness of the superpixels [80].

3.3.2. Method of SLIC

SLIC is easy to use and comprehend. The algorithm's sole parameter by default is k or the required number of about equal-sized superpixels. The clustering process starts with an initialization stage where k initial cluster centers $C_i = \begin{bmatrix} l_i & a_i & b_i x_i & y_i \end{bmatrix}^{\mathrm{T}}$ are sampled on a regular grid spaced S pixels apart for color images in the CIELAB color space. The grid interval is set to $S = \sqrt{N/k}$ to create superpixels that are rough of identical size. The centers are shifted to the seed positions that correspond to a 3×3 neighborhood's lowest gradient position. This prevents a superpixel from being centered on an edge and lessens the possibility of seeding a superpixel with a noisy pixel [80].

Following that, each pixel i is assigned to the closest cluster center that overlaps the search region located in the assignment stage. This is essential for accelerating the algorithm because reducing the search region's size results in a large speedup. Compared to classical k-means clustering, where each pixel must be compared to all cluster centers, the quantity and outcomes of distance calculations offer a substantial time benefit. Only the addition of the distance measure D, which identifies the closest cluster center for each pixel, makes this conceivable. The search for similar pixels is conducted in the $2S \times 2S$ region surrounding the superpixel center because the expected spatial size of a superpixel is a region of about $S \times S$ size [80].

After assigning each pixel to the closest cluster center, an update step changes the cluster centers such that they represent the average $\begin{bmatrix} l & b & x & y \end{bmatrix}^{\mathrm{T}}$ a vector containing every pixel in the cluster. The residual E error between the new cluster center sites and the old cluster center locations is calculated using the L_2 norm. Up until the error converges, the assignment and update processes can be iteratively repeated. The connection is then forced in a post-processing phase by reassigning specific pixels to neighboring superpixels [80].

Clusters in the *labxy* color image plane space correspond to SLIC superpixels. This creates a challenge in defining the distance measure D that might not be evident right away. D figures out how far a pixel i is from the cluster center C_k. The CIELAB color space $[lab]^T$, whose possible range of values is known, is used to represent the color of a pixel. The possible values for a pixel whose position is $\begin{bmatrix} x & y \end{bmatrix}^{\mathrm{T}}$ depend on the size of the image [80].

Different superpixel sizes will exhibit inconsistent clustering behavior if D is defined as 5D euclidean distances in *labxy* space. Large superpixels favor spatial proximity over color affinity and favor spatial distance over color. Compact superpixels are created as a result, although they do not cling to image borders well. Superpixels with a lesser size work the opposite way [80].

It is important to normalize the color affinity and spatial proximity to the corresponding maximum distances N_s and N_c within a cluster in order to integrate the two distances into a single measure. By doing this, D' is equal to,

$$d_c = \sqrt{\left(l_j - l_i\right)^2 + \left(a_j - a_i\right)^2 + \left(b_j - b_i\right)^2}, \tag{3.6}$$

$$d_s = \sqrt{\left(x_j - x_i\right)^2 + \left(y_j - y_i\right)^2}, \tag{3.7}$$

$$D' = \sqrt{\left(\frac{d_c}{N_c}\right)^2 + \left(\frac{d_s}{N_s}\right)^2} \tag{3.8}$$

Within a given cluster, the expected maximum spatial distance should correspond to the sampling interval $N_S = S = \sqrt{(N/K)}$. The greatest color distance N_c can vary greatly from cluster to cluster and from image to image, making it difficult to determine. Fixing N_c to a fixed value of m will avoid this problem, so

$$D' = \sqrt{\left(\frac{d_c}{m}\right)^2 + \left(\frac{d_s}{S}\right)^2} \tag{3.9}$$

simplified:

$$D = \sqrt{d_c^2 + \left(\frac{d_s}{S}\right)^2 m^2} \tag{3.10}$$

By defining D in this way, m also enables us to compare the relative weights of color similarity and spatial closeness. When m is bigger, superpixels will be more compact. When m is smaller, superpixels will tightly stick to image boundaries but are less symmetrical in size and form [80].

3.4. Segmentations Made With The Superpixels Method

Before moving on to the proposed method, I would like to summarize the other methods of DGL testing with superpixels. I'll also explain how I used these other methods to my advantage and how I improved my method.

3.4.1. Adding superpixels to the DGL test

Let $\mathscr{T}_1, \mathscr{T}_2, \ldots, \mathscr{T}_M$ be M sets of image pixels with user labels. The ground truth masks and bounding boxes, which we discuss in more detail in the following section, can be used to obtain these pixels. Let $\hat{Q}_i : \mathscr{X} \to [0,1]$, $\mathscr{X} = \mathscr{X}_1 \times \mathscr{X}_2 \times \ldots \times \mathscr{X}_d, i = 1,2,\ldots,M$, represent the potential $\mathscr{T}_i$ quantized intensity histograms. As a formula,

$$\hat{Q}_i(q)_{q\in\mathscr{X}} = \frac{1}{|\mathscr{T}_i|} \sum_{X\in\mathscr{T}_i} \mathbb{I}_{\bar{I}(X)=q} \tag{3.11}$$

where $\mathbb{I}$ is an indicator function and $\bar{I}(X)$ stands for the rounded intensity of pixel X; the pixel's intensity is rounded toward the histogram bin descriptor's nearest edge.

By partitioning $\Omega \rightarrow \{\mathscr{S}_1, \mathscr{S}_2, \ldots, \mathscr{S}_K\}$, $K \leqslant N$ first, and then applying the DGL test to each superpixel, we perform picture labeling at the superpixel level. Superpixels' border adherences are their central tenet, however in practice, this may not always be the case, and applying picture labels at the superpixel level may lead to some residual mistakes. However, because they combine adjacent pixels into perceptually identical cells that are simple to distinguish using the DGL test, their adoption simplifies the technique.

One must compute $M(M-1)/2$ Borel set $A_{i,j}, A_{i,j} \in \mathscr{A}$, in order to use the DGL test.

$$A_{i,j} = \left\{q : \hat{Q}_i(q) \geqslant \hat{Q}_j(q)\right\}, \quad 1 \leqslant i < j \leqslant M, \tag{3.12}$$

for all $q \in \mathscr{X}$. Then, if we let $\{X_1, X_2, \ldots, X_n\}$ be the set of pixels in $\mathscr{S}_k$, the DGL test decides $\hat{L}(\mathscr{S}_k) = i$ if

$$\max_{A \in \mathscr{A}} \left| \int_A \hat{Q}_i(A) - \mu_n(A) \right| = \min_{j=1,\ldots,M} \max_{A \in \mathscr{A}} \left| \int_A \hat{Q}_j(A) - \mu_n(A) \right| \tag{3.13}$$

where $\mathscr{X}$ can be used to numerically calculate $\mu_n(A) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{I}_{X_i \in A}$ and the integrals in 3.14 [81].

3.4.2. DGL test with the centroid of labeled pixels and superpixels

Noticed that the baseline method's decision rule in 3.14 intends to select the hypothesis with the least amount of mismatch between the empirical and nominal histograms. Therefore, this method employs the relative distances between the intensity histograms of the input regions rather than the geographical properties of the user's input regions. However, these regions' spatial data may also be very important for the segmentation process. For example, if a user provides a region's bounding box, the region of interest will be contained within that bounding box, so one shouldn't hunt for it elsewhere in the image.

The DGL can incorporate the spatial information of the user input regions by creating spatially variable intensity distributions [32]. In this method, the alphabet of the intensity distributions, $\mathscr{X}$, is added to the spatial dimension, or the pixel grid Ω, while also taking into account a combination alphabet, C $= \mathscr{X} \times \Omega$. When using this alphabet, the probability of a pixel's intensity, I(X), is taken into account along with its spatial location, X, as $\Pr(I(X),X)$. As a result, both in the histogram and the spatial domain, the resulting spatially variable distributions allow discrimination. The results of the Borel sets computations A $\in$ in 3.13 and the implementation of the test in 3.14 must be carried out over C rather than X This needs to be modified approach to the DGL.

The proposed approach for this method is implemented as follows. $L_1, L_2, \ldots . L_M$ labeled pixels collected from the periphery of user input zones are used.

$$C_j = \frac{1}{|L_i|} \sum_{x \in L_i} X \tag{3.14}$$

Let L_i be centroid of the pixels:

$$\mu\left(S_k\right) = \frac{1}{|S_k|} \sum_{x \in S_k} X \tag{3.15}$$

And S_k is the centroid of the superpixel. The baseline technique seeks to minimize the measure $\max_{A \in \mathscr{A}} \left| \int_A \hat{Q}_j(A) - \mu_n(A) \right|$ in the color domain when choosing a label for S_k.

This measure shows the difference between the support's empirical and nominal distributions $A \in \mathscr{A}$. According to an analogy, the term $\left|C_j - \mu\left(S_k\right)\right|$ refers to the mismatch between the nominal position of the area Ωj and the centroid of the superpixel.

$$i = \quad \arg \min_{j=1,\ldots,M} \max_{A \in \mathscr{A}} \left| \int_A \hat{Q}_j(A) - \mu_n(A) \right| + w_j \left|C_j - \mu\left(S_k\right)\right| \tag{3.16}$$

where the weight terms, $w_j \geq 0$, can be arranged to alter the degree to which a mismatch predominates in either the color or spatial domains. The suggested test becomes equal to the baseline test as $w_j \to 0$, and the proposed spatial mismatch term's effect disappears. While the test becomes simply spatial as $w_j \to \infty$, the impact of the color mismatch is no longer present. As a result, we balance the two mismatches by carefully choosing the w_j terms. The suggested algorithm was applied by adjusting $w_j = 1$ [82].

3.5. Spatially Varying Color Distribution Based DGL Test Segmentation

The method described above reduces, but not completely, errors outside the bounding boxes. Starting from this method, I developed the method further and passed the errors outside the bounding boxes, and the accuracy of image segmentation increased. In this section, I explained in detail how I developed it.

$$A_{i,j} = \left\{x : \hat{Q}_i(x) \geqslant \hat{Q}_j(x)\right\}, \quad 1 \leqslant i < j \leqslant M, \tag{3.17}$$

We write $\hat{Q}_i(x,z)$ for $\hat{Q}_i(x)$ and $\hat{Q}_j(x,z)$ for $\hat{Q}_j(x)$. i and j represent the distributions here and x are the two-dimensional H,S values. z is a random variable that gives us information about the centroid of superpixels in BB_i or not in. That is, it indicates spatial information. And probability formula for $\hat{Q}_i(x,z)$ is equal to (it means probability both occur together):

$$\hat{Q}_i(x,z) = \hat{Q}_i(x|z) * \hat{Q}_i(z) \tag{3.18}$$

This time we are faced with two situations.

- z=1: If superpixels centorid in BB_i
- z=0: If superpixels centorid not in BB_i

In the proposed method, we did not adapt the relative sizes of the Bounding Boxes to the method. That is, we did not take into account the prior probability of $\hat{Q}_i(z)$. We only operate for $\hat{Q}_i(x|z)$, so there are two cases.

$$\hat{Q}_i(x|z) = \begin{cases} \hat{Q}_i(x), & \text{if } z = 1 \\ 0, & \text{otherwise} \end{cases} \tag{3.19}$$

$$A_{i,j} = \left\{x : \hat{Q}_i(x,z) \geqslant \hat{Q}_j(x,z)\right\} = \left\{x : \hat{Q}_i(x|z) * \hat{Q}_i(z) \geqslant \hat{Q}_j(x|z) * \hat{Q}_j(z)\right\} \tag{3.20}$$

If we start from here, it turns out that we have four situations. But when $z \notin BB_i$, it will be an empty set because it going to be zero. Therefore, only two cases were considered.

For the first case: $z \notin BB_i$ and $z \in BB_j$

$$A_1 = \left\{x : 0 \geqslant \hat{Q}_j(x) * \hat{Q}_j(z)\right\}, \quad 1 \leqslant i < j \leqslant M, \tag{3.21}$$

For the second case: $z \notin BB_i$ and $z \notin BB_j$

$$A_2 = \{x : 0 \geqslant 0\}, \quad 1 \leqslant i < j \leqslant M, \tag{3.22}$$

Since the left of the equality is zero in the above two cases, it cannot enter inside the bounding boxes anyway and goes to other cases.

For the third case: $z \in BB_i$ and $z \in BB_j$

$$A_1 = \left\{x : \hat{Q}_i(x) * \hat{Q}_i(z) \geqslant \hat{Q}_j(x) * \hat{Q}_j(z)\right\}, \quad 1 \leqslant i < j \leqslant M, \tag{3.23}$$

For the fourth case: $z \in BB_i$ and $z \notin BB_j$

$$A_2 = \{x : \hat{Q}_i(x) * \hat{Q}_i(z) \geqslant 0\}, \quad 1 \leqslant i < j \leqslant M, \tag{3.24}$$

Distribution sets i and j of X were created above. After the clusters are created, we calculate their probabilities from the M distribution.

$$\max_{A \in \mathscr{A}} \left| \int_A \hat{Q}_i(A) - \mu_n(A) \right| = \min_{j=1,\ldots,M} \max_{A \in \mathscr{A}} \left| \int_A \hat{Q}_j(A) - \mu_n(A) \right| \tag{3.25}$$

In other words, we take the integral over A and add the probabilities. When adding, we add from M. Likewise, M may or may not be an element of BB_i. As soon as they enter the BB_i, their status is also checked. μ_n is calculated if i and j are inside BB_i. If M also enters, M is not zero, this time $\int_A \hat{Q}_i(A)$ is chosen. $\int_A \hat{Q}_i(A)$ is already calculated from A_1. If M is not entered, $\int_A \hat{Q}_i(A) = 0$.

If i is entered and j is not entered, μ_n is calculated from A_2. If M is entered, A_2 is calculated. If M is not entered, it is already zero.

If i and j are not entered, $\mu_n = 0$ and $\int_A \hat{Q}_i(A) = 0$.

In the end, the maximum is chosen twice for both i and j. Finally, we print and sort, and choose the minimum. That is, when calculating errors, we take the minimum.

4. RESULTS AND DISCUSSIONS

The performances of the proposed methods are tested with simulations and then these methods are compared with each other. These simulations are presented in two sections, which are Sliding Windows and Spatially Varying techniques.

4.1. Sliding Window DGL Based Segmentation

The Berkeley Segmentation Data Set and Benchmarks provided the data sets that we used (BSDS500). The accuracy of the segmented photos from Berkeley's GT below was evaluated using our suggested techniques. You can understand how the soft approach functions by comparing the soft and hard methods in Fig. 3.2. The Soft DGL method produces somewhat smoother and better results than the Hard DGL method. Examples from the Berkeley dataset are displayed in Fig.3.3. There are some reasons why this method works as intended. So the disadvantages of this method. The first disadvantage is the roughness of the image. This is because we're shifting it frame by frame, so the result doesn't look nice and smooth. The second disadvantage is the lack of spatial information in this method. In our later methods, these two disadvantages were eliminated and better results were obtained.

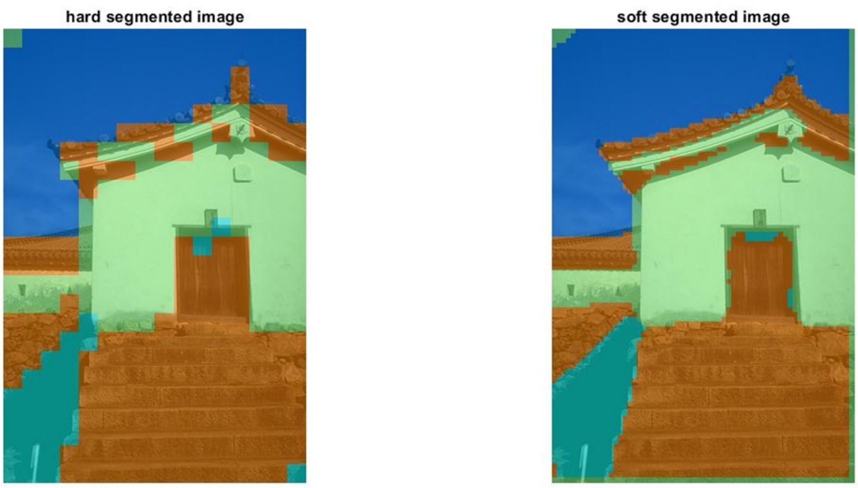

Figure 4.1: Both the first and second results are from different DGL tests 84% for Hard DGL accuracy and 85% for Soft DGL accuracy.

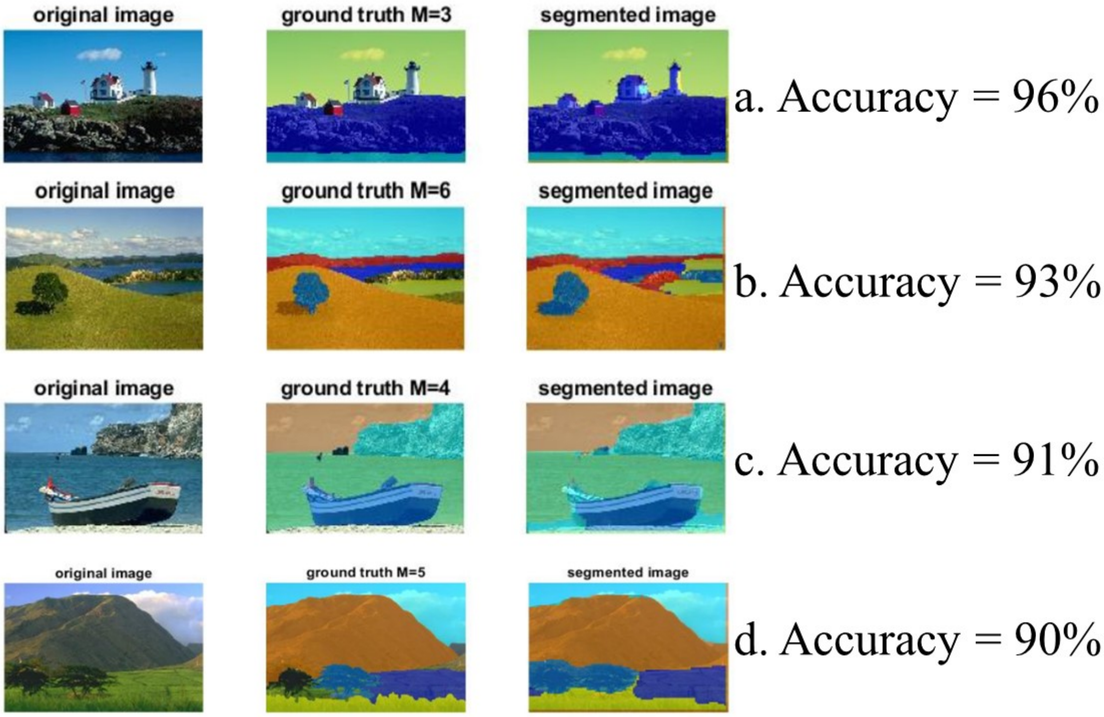

Figure 4.2: Using the Soft DGL approach, certain proposed method findings were accurately compared to the ground truth in the Berkeley dataset.

4.2. Spatially Varying Color Distribution Based DGL Test Segmentation

The effectiveness of the suggested method and the alternative methods [81] and [82] of the DGL test were compared. The simulations are run using test images from Berkeley's BSDS500 database [83]. 200 natural RGB photos are part of this collection, and each image has numerous GroundTruths annotations. Each color intensity is accurately encoded with 8 bits, or $|X| = 256^3$, in the 321×481 sized pixel grid. Below is an example segmented image from Berkeley's first GroundTruths that compares our suggested approach to various DGL Test approaches and measures accuracy.

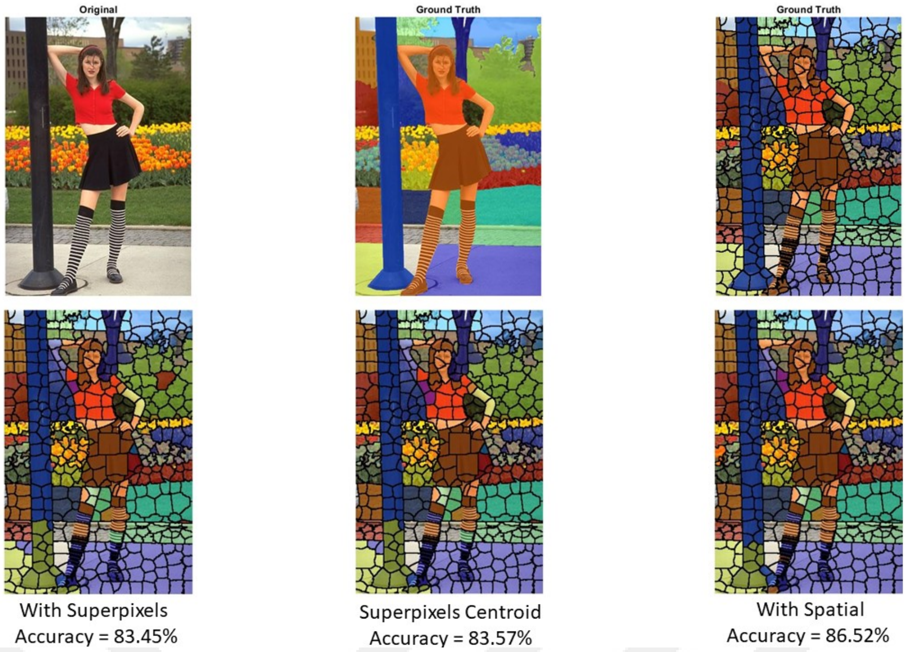

Figure 4.3: Original Image and its Ground Truth with and without superpixels and different segmentation methods' results with accuracies. The Spatial one is our proposed method.

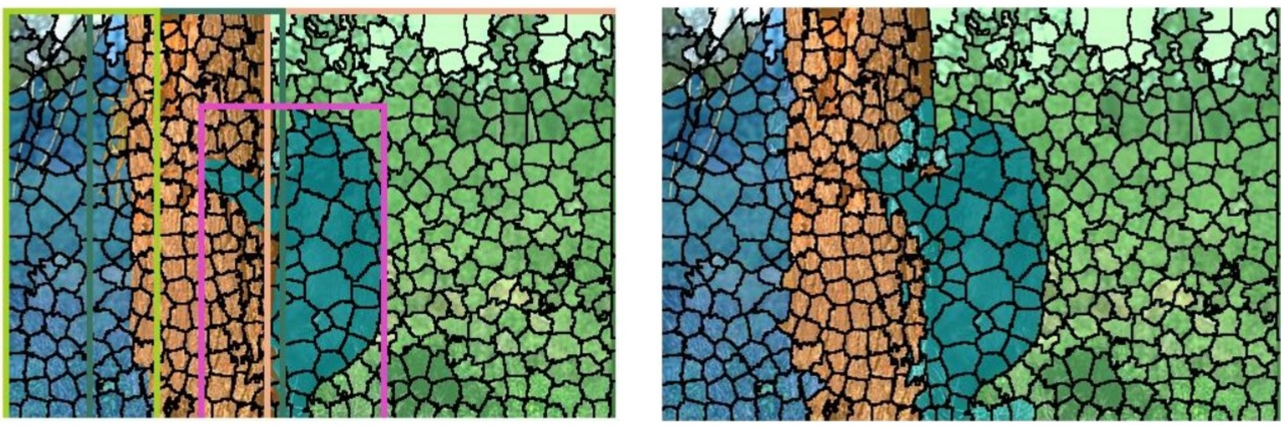

Figure 4.4: Example image for regions automatically selected by bounding boxes. You can see different colored bounding boxes.

4.2.1. Simulation of Superpixels and Spatially Varying Methods

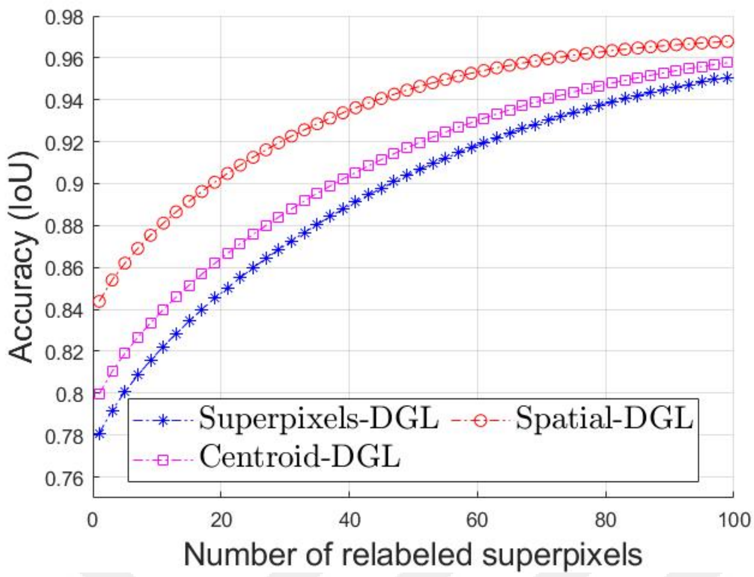

Figure 4.5: Results from comparison tests using Berkeley's BSDS500 database for the proposed Spatio-temporal (Spatial-DGL) DGL test and the other DGL test methodologies. The red curves show the proposed approach, the purple curves indicate the method using a centroid of pixels, and the blue curves represent the way of adding superpixels in all of the figures.

We have employed the intersection over union (IoU) metric, which is defined as:

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$

where the overlap and union of the ground truths and the superpixels in question are determined. By averaging it among several GroundTruths and images, IoU is calculated. We reduced the number of regions in each image and considered 90% of the labeled pixels before computing IoU because the bulk of the annotations are over-segmented, as in [81].

For each method that was taken into consideration, we compared how well all techniques performed using the other methods in [81] and [82]. The ground truth masks and bounding boxes of the image segments are denoted using the same notation as GT_i and BB_i, where $i = 1, 2, \ldots, M$.

All algorithms are implemented in the HSV color space, which only considers the H and S parts of the image.

By choosing $|\mathscr{X}_1| = |\mathscr{X}_2| = \sqrt{321 \times 481} \approx 392$ to achieve linear complexity in the number of pixels for all tests, we have implemented dimensionality reduction as in [81]. The majority of the images in the BSDS500 database have improved in accuracy as a result of this choice.

5. CONCLUSIONS

The successful and appropriate technique for image segmentation is color and spatial-based segmentation. For application in the user-assisted multi-label image segmentation problem, we provided a spatially adaptable variant of the DGL test. In order to improve segmentation performance, the suggested method offers a straightforward mechanism to include spatial information from user inputs into fundamental, color space-based DGL testing. We have demonstrated that the proposed method may be easily added to the existing ones, improving performance without adding to its complexity. Additionally, we offered benchmark results from Berkeley's BSDS500 database, demonstrating that an accuracy improvement of around 8% over the first (adding superpixels) and 5% over the second (centroid of pixels) techniques may be made. Image segmentation using the DGL approach will be automatic and user-defined. The suggested approach is multi-label and semi-supervised.

6. RECOMMENDATIONS

For future studies, have the plan to get better results by including the relative probabilities of bounding boxes. When describing the spatio-temporal method above, I stated that I did not include the marginal probability of bounding boxes. By adding the prior probability of $\hat{Q}_i(z)$ to the calculation, can get better and more punctual results by going beyond the last method we have done.

REFERENCES

1. Sharif, M., S. Mohsin, M. J. Jamal and M. Raza, “Illumination normalization preprocessing for face recognition”, *2010 The 2nd Conference on Environmental Science and Information Application Technology*, Vol. 2, pp. 44–47, IEEE, 2010.

2. Yasmin, M., M. Sharif, S. Masood, M. Raza and S. Mohsin, “Brain image enhancement-A survey”, *World Applied Sciences Journal*, Vol. 17, No. 9, pp. 1192–1204, 2012.

3. Seerha, G. K. and R. Kaur, “Review on recent image segmentation techniques”, *International Journal on Computer Science and Engineering*, Vol. 5, No. 2, p. 109, 2013.

4. Acharya, J., S. Gadhiya and K. Raviya, “Segmentation techniques for image analysis: A review”, *International Journal of computer science and management research*, Vol. 2, No. 1, pp. 1218–1221, 2013.

5. Hameed, M., M. Sharif, M. Raza, S. W. Haider and M. Iqbal, “Framework for the comparison of classifiers for medical image segmentation with transform and moment based features”, *Research Journal of Recent Sciences*, Vol. 2277, p. 2502, 2012.

6. Yasmin, M., M. Sharif, S. Masood, M. Raza and S. Mohsin, “Brain image reconstruction: A short survey”, *World Applied Sciences Journal*, Vol. 19, No. 1, pp. 52–62, 2012.

7. Khan, W., “Image segmentation techniques: A survey”, *Journal of Image and Graphics*, Vol. 1, No. 4, pp. 166–170, 2013.

8. Kohli, P., A. Osokin and S. Jegelka, “A principled deep random field model for image segmentation”, *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1971–1978, 2013.

9. Singh, P. and R. S. Chadha, “A novel approach to image segmentation”, *International Journal of Advanced Research in Computer Science and Software Engineering*, Vol. 3, No. 4,

2013.

10. Kaganami, H. G. and Z. Beiji, “Region-based segmentation versus edge detection”, *2009 Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pp. 1217–1221, IEEE, 2009.

11. Sharma, N., M. Mishra and M. Shrivastava, “Colour image segmentation techniques and issues: an approach”, *International Journal of Scientific & Technology Research*, Vol. 1, No. 4, pp. 9–12, 2012.

12. Dass, R. and S. Devi, “Image segmentation techniques 1”, , 2012.

13. Ning, J., L. Zhang, D. Zhang and C. Wu, “Interactive image segmentation by maximal similarity based region merging”, *Pattern Recognition*, Vol. 43, No. 2, pp. 445–456, 2010.

14. Zaitoun, N. M. and M. J. Aqel, “Survey on image segmentation techniques”, *Procedia Computer Science*, Vol. 65, pp. 797–806, 2015.

15. Sharif, M., M. A. Ali, M. Raza and S. Mohsin, “Face recognition using edge information and DCT”, *Sindh University Research Journal-SURJ (Science Series)*, Vol. 43, No. 2, 2015.

16. Lakshmi, S., D. V. Sankaranarayanan *et al.*, “A study of edge detection techniques for segmentation computing approaches”, *IJCA Special Issue on “Computer Aided Soft Computing Techniques for Imaging and Biomedical Applications” CASCT*, pp. 35–40, 2010.

17. Hussain, Z. and D. Agarwal, “A comparative analysis of edge detection techniques used in flame image processing”, *International Journal of Advance Research In Science And Engineering IJARSE*, , No. 4, 2015.

18. Raja, S. K., A. S. ABDUL KHADIR and S. R. Ahamed, “MOVING TOWARD REGION-BASED IMAGE SEGMENTATION TECHNIQUES: A STUDY.”, *Journal of Theoretical & Applied Information Technology*, Vol. 5, No. 1, 2009.

19. Vicente, S., V. Kolmogorov and C. Rother, “Graph cut based image segmentation with

connectivity priors", *2008 IEEE conference on computer vision and pattern recognition*, pp. 1–8, IEEE, 2008.

20. Li, Y., J. Sun, C.-K. Tang and H.-Y. Shum, "Lazy snapping", *ACM Transactions on Graphics (ToG)*, Vol. 23, No. 3, pp. 303–308, 2004.

21. Yi, F. and I. Moon, "Image segmentation: A survey of graph-cut methods", *2012 international conference on systems and informatics (ICSAI2012)*, pp. 1936–1941, IEEE, 2012.

22. Boykov, Y. and G. Funka-Lea, "Graph cuts and efficient ND image segmentation", *International journal of computer vision*, Vol. 70, No. 2, pp. 109–131, 2006.

23. Van Der Zwaag, B. J., C. Slump and L. Spaanenburg, "Analysis of neural networks for edge detection", *Proceedings of the ProRISC Workshop on Circuits, Systems and Signal Processing*, pp. 580–586, 2002.

24. SowmiyaNarayanan, G., "Image Segmentation Using Mask R-CNN.", , 2020.

25. Adams, R. and L. Bischof, "Seeded region growing", *IEEE Transactions on pattern analysis and machine intelligence*, Vol. 16, No. 6, pp. 641–647, 1994.

26. Vezhnevets, V. and V. Konouchine, "GrowCut: Interactive multi-label ND image segmentation by cellular automata", *proc. of Graphicon*, Vol. 1, pp. 150–156, Citeseer, 2005.

27. Meena, S., K. Palaniappan and G. Seetharaman, "User driven sparse point-based image segmentation", *2016 IEEE International Conference on Image Processing (ICIP)*, pp. 844–848, IEEE, 2016.

28. Song, G., H. Myeong and K. M. Lee, "Seednet: Automatic seed generation with deep reinforcement learning for robust interactive segmentation", *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1760–1768, 2018.

29. Mahadevan, S., P. Voigtlaender and B. Leibe, "Iteratively trained interactive segmentation", *arXiv preprint arXiv:1805.04398*, 2018.

30. Xu, N., B. Price, S. Cohen, J. Yang and T. S. Huang, "Deep interactive object selection", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 373–381, 2016.

31. Friedland, G., K. Jantz and R. Rojas, "SIOX: Simple interactive object extraction in still images", *Seventh IEEE International Symposium on Multimedia (ISM'05)*, pp. 7–pp, IEEE, 2005.

32. Nieuwenhuis, C. and D. Cremers, "Spatially varying color distributions for interactive multilabel segmentation", *IEEE transactions on pattern analysis and machine intelligence*, Vol. 35, No. 5, pp. 1234–1247, 2012.

33. Xiang, S., F. Nie, C. Zhang and C. Zhang, "Interactive natural image segmentation via spline regression", *IEEE transactions on image processing*, Vol. 18, No. 7, pp. 1623–1632, 2009.

34. Long, J., X. Feng, X. Zhu, J. Zhang and G. Gou, "Efficient superpixel-guided interactive image segmentation based on graph theory", *Symmetry*, Vol. 10, No. 5, p. 169, 2018.

35. Duchenne, O., J.-Y. Audibert, R. Keriven, J. Ponce and F. Ségonne, "Segmentation by transduction", *2008 IEEE conference on computer vision and pattern recognition*, pp. 1–8, IEEE, 2008.

36. Meshry, M., A. Taha and M. Torki, "Multi-Modality Feature Transform: An Interactive Image Segmentation Approach.", *BMVC*, pp. 72–1, Citeseer, 2015.

37. Ren, Y., C.-S. Chua and Y.-K. Ho, "Statistical background modeling for non-stationary camera", *Pattern Recognition Letters*, Vol. 24, No. 1-3, pp. 183–196, 2003.

38. Kim, T. H., K. M. Lee and S. U. Lee, "Nonparametric higher-order learning for interactive segmentation", *2010 IEEE computer society conference on computer vision and pattern recognition*, pp. 3201–3208, IEEE, 2010.

39. Bai, J. and X. Wu, "Error-tolerant scribbles based interactive image segmentation", *Pro-*

ceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 392–399, 2014.

40. Zhang, J., Z.-h. Tang, W.-h. Gui, Q. Chen and J.-p. Liu, "Interactive image segmentation with a regression based ensemble learning paradigm", *Frontiers of Information Technology & Electronic Engineering*, Vol. 18, No. 7, pp. 1002–1020, 2017.

41. Subr, K., S. Paris, C. Soler and J. Kautz, "Accurate binary image selection from inaccurate user input", *Computer Graphics Forum*, Vol. 32, pp. 41–50, Wiley Online Library, 2013.

42. Kohli, P., P. H. Torr *et al.*, "Robust higher order potentials for enforcing label consistency", *International Journal of Computer Vision*, Vol. 82, No. 3, pp. 302–324, 2009.

43. Jian, M. and C. Jung, "Interactive image segmentation using adaptive constraint propagation", *IEEE transactions on image processing*, Vol. 25, No. 3, pp. 1301–1311, 2016.

44. Li, W., Y. Shi, W. Yang, H. Wang and Y. Gao, "Interactive image segmentation via cascaded metric learning", *2015 IEEE International Conference on Image Processing (ICIP)*, pp. 2900–2904, IEEE, 2015.

45. Luo, L., X. Wang, S. Hu, X. Hu and L. Chen, "Interactive image segmentation based on samples reconstruction and FLDA", *Journal of Visual Communication and Image Representation*, Vol. 43, pp. 138–151, 2017.

46. Taha, A. and M. Torki, "Seeded laplaican: An eigenfunction solution for scribble based interactive image segmentation", *arXiv preprint arXiv:1702.00882*, 2017.

47. Grady, L., "Random walks for image segmentation", *IEEE transactions on pattern analysis and machine intelligence*, Vol. 28, No. 11, pp. 1768–1783, 2006.

48. Wang, T., Z. Ji, Q. Sun, Q. Chen and S. Han, "Image segmentation based on weighting boundary information via graph cut", *Journal of Visual Communication and Image Representation*, Vol. 33, pp. 10–19, 2015.

49. Tang, M., D. Marin, I. B. Ayed and Y. Boykov, “Kernel Cuts: MRF meets kernel & spectral clustering”, *arXiv preprint arXiv:1506.07439*, 2015.

50. Bai, X. and G. Sapiro, “A geodesic framework for fast interactive image and video segmentation and matting”, *2007 IEEE 11th International Conference on Computer Vision*, pp. 1–8, IEEE, 2007.

51. Price, B. L., B. Morse and S. Cohen, “Geodesic graph cut for interactive image segmentation”, *2010 IEEE computer society conference on computer vision and pattern recognition*, pp. 3161–3168, IEEE, 2010.

52. Gong, Y., S. Xiang, L. Wang and C. Pan, “Fine-structured object segmentation via edge-guided graph cut with interaction simplification”, *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1801–1805, IEEE, 2016.

53. Casaca, W., L. Gustavo Nonato and G. Taubin, “Laplacian coordinates for seeded image segmentation”, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 384–391, 2014.

54. Li, Y., J. Sun, C.-K. Tang and H.-Y. Shum, “Lazy snapping”, *ACM Transactions on Graphics (ToG)*, Vol. 23, No. 3, pp. 303–308, 2004.

55. Wang, J., “Discriminative Gaussian mixtures for interactive image segmentation”, *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP’07*, Vol. 1, pp. I–601, IEEE, 2007.

56. Yang, W., J. Cai, J. Zheng and J. Luo, “User-friendly interactive image segmentation through unified combinatorial user inputs”, *IEEE Transactions on Image Processing*, Vol. 19, No. 9, pp. 2470–2479, 2010.

57. Shi, R., Z. Liu, Y. Xue and X. Zhang, “Interactive object segmentation using iterative adjustable graph cut”, *2011 Visual Communications and Image Processing (VCIP)*, pp. 1–4, IEEE, 2011.

58. Wang, T., Z. Ji, Q. Sun, Q. Chen, Q. Ge and J. Yang, "Diffusive likelihood for interactive image segmentation", *Pattern Recognition*, Vol. 79, pp. 440–451, 2018.

59. Peng, B., L. Zhang, D. Zhang and J. Yang, "Image segmentation by iterated region merging with localized graph cuts", *Pattern Recognition*, Vol. 44, No. 10-11, pp. 2527–2538, 2011.

60. Alain, M., C. Guillemot, D. Thoreau, P. Guillotel, R. Ammanouil, A. Ferrari, C. Richard, S. Mathieu, C. Bampis, Z. Li *et al.*, "An, D., see Chen, L., TIP Nov. 2017 5545-5554 Ancuti, C., see Ancuti, CO, TIP Jan. 2017 65-78 Ancuti, CO, Ancuti, C., De Vleeschouwer, C., and Bovik, AC, Single-Scale Fusion: An Effective Approach to Merging Images; TIP Jan. 2017 65-78", *IEEE Transactions on Image Processing*, Vol. 26, No. 12, p. 6097, 2017.

61. Zhang, J., J. Zheng and J. Cai, "A diffusion approach to seeded image segmentation", *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 2125–2132, IEEE, 2010.

62. Ducournau, A. and A. Bretto, "Random walks in directed hypergraphs and application to semi-supervised image segmentation", *Computer Vision and Image Understanding*, Vol. 120, pp. 91–102, 2014.

63. Tang, M., D. Marin, I. B. Ayed and Y. Boykov, "Normalized cut meets MRF", *European Conference on Computer Vision*, pp. 748–765, Springer, 2016.

64. Jegelka, S. and J. Bilmes, "Submodularity beyond submodular energies: coupling edges in graph cuts", *CVPR 2011*, pp. 1897–1904, IEEE, 2011.

65. Kohli, P., A. Osokin and S. Jegelka, "A principled deep random field model for image segmentation", *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1971–1978, 2013.

66. Nguyen, T. N. A., J. Cai, J. Zhang and J. Zheng, "Robust interactive image segmentation using convex active contours", *IEEE Transactions on Image Processing*, Vol. 21, No. 8, pp. 3734–3743, 2012.

67. Liu, D., Y. Xiong, L. Shapiro and K. Pulli, "Robust interactive image segmentation with automatic boundary refinement", *2010 IEEE International Conference on Image Processing*, pp. 225–228, IEEE, 2010.

68. Li, H., M. Gong, Q. Miao and B. Wang, "Interactive active contour with kernel descriptor", *Information Sciences*, Vol. 450, pp. 53–72, 2018.

69. Ning, J., L. Zhang, D. Zhang and C. Wu, "Interactive image segmentation by maximal similarity based region merging", *Pattern Recognition*, Vol. 43, No. 2, pp. 445–456, 2010.

70. Zhou, C., D. Wu, W. Qin and C. Liu, "An efficient two-stage region merging method for interactive image segmentation", *Computers & Electrical Engineering*, Vol. 54, pp. 220–229, 2016.

71. Rother, C., V. Kolmogorov and A. Blake, "" GrabCut" interactive foreground extraction using iterated graph cuts", *ACM transactions on graphics (TOG)*, Vol. 23, No. 3, pp. 309–314, 2004.

72. Tang, M., L. Gorelick, O. Veksler and Y. Boykov, "Grabcut in one cut", *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1769–1776, 2013.

73. Yu, H., Y. Zhou, H. Qian, M. Xian and S. Wang, "Loosecut: Interactive image segmentation with loosely bounded boxes", *2017 IEEE International Conference on Image Processing (ICIP)*, pp. 3335–3339, IEEE, 2017.

74. Cheng, M.-M., V. A. Prisacariu, S. Zheng, P. H. Torr and C. Rother, "Densecut: Densely connected crfs for realtime grabcut", *Computer Graphics Forum*, Vol. 34, pp. 193–201, Wiley Online Library, 2015.

75. Rajchl, M., M. C. Lee, O. Oktay, K. Kamnitsas, J. Passerat-Palmbach, W. Bai, M. Damodaram, M. A. Rutherford, J. V. Hajnal, B. Kainz *et al.*, "Deepcut: Object segmentation from bounding box annotations using convolutional neural networks", *IEEE transactions on medical imaging*, Vol. 36, No. 2, pp. 674–683, 2016.

76. Chen, Y., A. B. Chan and G. Wang, "Adaptive figure-ground classification", *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 654–661, IEEE, 2012.

77. Kay, S. M., *Fundamentals of statistical signal processing*, Prentice Hall, 1998.

78. Lehmann, E., *Testing statistical hypotheses*, Wiley series in probability and mathematical statistics: Probability and mathematical statistics. Wiley, 1986.

79. Devroye, L., L. Gyorfi and G. Lugosi, "A note on robust hypothesis testing", *IEEE Transactions on Information Theory*, Vol. 48, No. 7, pp. 2111–2114, 2002.

80. Achanta, R., A. Shaji, K. Smith, A. Lucchi, P. Fua and S. Süsstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods", *IEEE transactions on pattern analysis and machine intelligence*, Vol. 34, No. 11, pp. 2274–2282, 2012.

81. Afşer, H., "A Baseline Statistical Method for Robust User-Assisted Multiple Segmentation", *IEEE Signal Processing Letters*, Vol. 29, pp. 737–741, 2022.

82. Afşer, H., "Spatially Adaptive DGL Test for Robust User-Assisted Multilabel Segmentation", *Çukurova Üniversitesi Mühendislik Fakültesi Dergisi*, Vol. 37, No. 2, pp. 569–576.

83. Arbelaez, P., M. Maire, C. Fowlkes and J. Malik, "Contour detection and hierarchical image segmentation", *IEEE transactions on pattern analysis and machine intelligence*, Vol. 33, No. 5, pp. 898–916, 2010.