A Project thesis on Super pixel based Text Segmentation from Scene Images

M.Tech
(2018 - 2020)

Project Submitted By:-
Name: -**GAJI ABBAS BIN AMIR**

Roll no.:-CSE182017

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

ALIAH UNIVERSITY, NEWTOWN CAMPUS

Under the guidance of

Dr. AYATULLAH FARUK MOLLAH

ASST. PROF OF CSE, ALIAH UNIVERSITY

July 2020

Declaration

I hereby declare that the dissertation "**A Project thesis on Superpixel based Text Segmentation from Scene Images**" submitted by me to the ***Department of Computer Science and Engineering, Aliah University***, Newtown, Kolkata - 700160 in partial fulfillment of the requirements for the award of ***Master of Technology*** in ***Computer Science and Engineering*** is a bona-fide record of the work carried out by me under the supervision of **Dr. Ayatullah Faruk Mollah**

I further declare that the work reported in this dissertation, has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma of this institute or of any other institute or University.

(Signature)

GAJI ABBAS BIN AMIR

Date:

Examination Roll No: CSE182017

University Reg. No: CSE144043 of 2014-15

Department of Computer Science and Engineering

Certificate

This is to certify that the dissertation entitled "**A Project thesis on Superpixel based Text Segmentation from Scene Images**" submitted by **Gaji Abbas Bin Amir** (Roll No: CSE182017, Reg. No: CSE144043 of 2014-15) to ***Aliah University***, in partial fulfillment of the requirement for the award of the degree of ***Master of Technology*** in ***Computer Science and Engineering*** is a bona-fide work carried out under my supervision. The dissertation fulfills the requirements as per the regulations of this University and in my opinion meets the necessary standards for submission. The contents of this dissertation have not been submitted and will not be submitted either in part or in full, for the award of any other degree or diploma and the same is certified.

(Signature of the Supervisor)
Dr. Ayatullah Faruk Mollah
Assistant Professor
Department of CSE, Aliah University

(Signature of Head of the Department)
Dr. Sk Md Obaidullah
Associate Professor and Head
Department of CSE, Aliah University

(Seal of the Department)
Date:
Place: Aliah University, Kolkata

Abstract

Scene text detection is a crucial step in end-to-end scene text recognition, a greatly challenging problem in computer vision. This paper proposes a novel scene text detection method that involves superpixel based stroke feature transform (SSFT) and deep learning based region classification (DLRC). Which consists in partitioning an input image into several regions via superpixel based clustering, removing most regions based on predefined criteria satisfied by the characters, and refining the remaining regions to obtain CCRs by computing a stroke width map. The character regions are identified from the CCRs using DLRC, in which several hand-crafted low-level features, i.e. Colour, texture.

Acknowledgements

In the process of making this project report a reality, I have been honoured with some valuable assistance from various people. I am deeply indebted to many people who have contributed in making the completion of this project work possible.

It is with heartfelt respect that I acknowledge my gratitude to **Dr. Ayatullah Faruk Mollah**, Assistant Professor, Department of Computer Science and Engineering, Aliah University, for his invaluable guidance and supervision during the course of my project work.

I express my sincere appreciation to **DR. SK MD OBAIDULLAH**, Head of the Department, Computer Science and Engineering Department, Aliah University, for making available all the facilities. I am also thankful to our other distinguished faculty members who have played a vital role in bringing us to the level of engineering standards.

I would like to offer my special thanks to SK ASIF ALI,MCA,ALIAH UNIVERSITY for his technical support on my study.I owe my honesty, sincerity and perseverance to my family, without whose support and inspiration I would never have, come this far. Last but not the least is my friends, classmates and other well-wishers who have motivated me throughout my student life.

Gaji Abbas Bin Amir

Roll No: CSE182017

Reg. No: CSE144043 of 2014-15

Content

Page No.

Dedicated to My Parents and Teachers

Chapter 1

Introduction

1.1 Introduction to Image Segmentation

In image processing, image segmentation is considered as a very important and challenging process. The process of partitioning an image into essential parts that contain similar features and properties is called image segmentation. Simplification is the main objective of segmentation. This process represents an image into significant way. The image obtained from segmentation can be analysed easily. In image analysis, image segmentation step plays an important role. Dividing an image into various regions or segments with similar features or attributes is the main aim of image segmentation.

a. Discontinuity detection based approach: In this approach, the segmentation of an image is done into regions on the basis of discontinuity. The edge detection based segmentation is a perfect example of this approach. This approach detects edges created because of the discontinuity in intensity. These edges are connected to create edges of regions.

b. Similarity detection based approach: In this approach, the image is segmented into several regions on the basis of similarity. The examples of this approach include thresholding, region growing methods and region splitting and merging. All these approaches partition an image into regions that comprise similar set of pixels. This approach is also utilized by clustering algorithms. On basis of some predefined criterion, these algorithms partition an image into a set of clusters that include similar features [2]

Image segmentation can be performed using large numbers of available methodologies. Every method has its unique significance. All of the existing methods are based on two fundamental segmentation approaches. These approaches include region based or edge based segmentation. Different segmentation methods are implemented on different images for carrying out segmentation task. There are mainly three categories in which the classification of segmentation methods can be done. These are:

i. Structural Segmentation Techniques: The structural segmentation techniques are based on the structure that contains information of the region of interest. This means that these techniques contain the information of the region selected for segmentation [3].

ii. Stochastic Segmentation Techniques: In place of working on the structural information of the region, stochastic segmentation techniques works on the discrete pixel values of the image.

iii. Hybrid Techniques: The hybrid techniques of image segmentation are based on the idea of both of the above mentioned methods. These techniques collectively use discrete pixel and structural information.

1.2 Image Segmentation Techniques

A brief overview of different image segmentation techniques is provided below:

a. Thresholding Method: Thresholding techniques are the most simple techniques of image segmentation. These techniques perform the division of image pixels on the basis of their brightness value. These methods use over images. In contrast to background, these types of images contain lighter objects. These techniques can be selected automatically or manually. The prior knowledge or information of image features can be used for selection purpose [4].

i. Local Threshold: Local threshold is based on the standard gray value and intensity value of input image. This approach partitions an input image into various sub-regions. Also, a different threshold value is selected for every sub region.

ii. Global Threshold: Global threshold is based on just gray level values. The threshold value completely belongs to the pixel's quality.

b. Edge Based Segmentation Method: Edge detection is a very popular field within image processing. Edge detection is basically an image segmentation technique. The edge based segmentation generally depends on the quick change in intensity value within an image. This is due to the reason that it is not possible for a single intensity value to provide good knowledge of edges. Edge detection methods find the edges in such a scenario when either the first derivative of intensity goes beyond a specific threshold or the second derivative contains zero crossings [5]. Edge detection is the first step in edge based segmentation method. Afterward, these detected edges are connected together for creating the object's edges. This is done for the segmentation of required regions. Gray histograms and Gradient based methods are the two fundamental edge based segmentation methods. It is possible to use any of the edge detection methods for edge detection. These methods include sobel operator, canny operator and Robert's operator and so on. These techniques generally provide binary image. These are the structural methods that depend on the identification of discontinuity.

C.Text segmentation using super pixel clustering

Super pixel generation

The experiments in [20] show that the SLIC approach has advantages in its adherence to boundaries; computational and memory efficiency; ease of use; ability to control super pixel compactness, regularity, and number. An adaptive SLIC text superpixels algorithm is proposed to generate super pixels for text images. Adaptive SLIC text super pixels: The SLIC superpixel approach groups pixels based on their colour similarity and proximity in the image plane by considering a five-dimensional (5D) space [L, a, b, x, y], where [L, a, b] represent three colour channels in CIELAB colour space and [x, y] is the pixel position in the image. SLIC utilises adaptive k-means clustering to produce superpixels. By initialising with the desired superpixel number K, the SLIC algorithm defines the ith initial cluster centre $S_i = [L_i, a_i, b_i, x_i, y_i]^T$, where $0 \le i \le K$, which forms regular, equally sized superpixel grid. To generate superpixels with similar sizes, the grid interval s is set to N/K, where N is the number of pixels. Then, each pixel is assigned to the nearest cluster centre. Finally, the cluster centres are updated iteratively according to the mean vector $[L, a, b, x, y]^T$ of all the pixels belonging to the cluster until convergence. The distance measure is calculated as

$$D = \mathrm{sqrt}(d^2c + (ds/s)^2 \cdot m^{2)}$$
$$d_c = \mathrm{sqrt}((L - Li)^2 + (a - ai)^2 + (b - bi)^2)$$
$$ds = (x - xi)^2 + (y - yi)^2 \qquad (1)$$

Where dc and ds are measurements of color similarity and spatial proximity, respectively, and m controls the compactness of superpixel clusters. In conventional SLIC algorithm, fixed parameters set by empirically cannot be suitable to all images, especially for text images. Taking characteristics of text into account, we proposed adaptive superpixel size and compactness to optimize superpixel generation for text images. Adaptive superpixel size: Fig. 3 shows SLIC superpixel images with different numbers of superpixels. The size of the original image is 235 × 63. Inappropriate superpixel size leads to adhered text strokes or inaccurate clusters for text and background. The boundaries between text and background were inaccurate when K = 100. Several stroke components were wrongly added into background superpixels, as shown in Fig. 3. When K = 300 or 500, the boundaries are distinct, but making the superpixels too small increases their sensitivity to noise. Owing to SW coherence, a superpixel size close to the stroke width is more likely to generate accurate boundaries between the text and background. Thus, we adjust the number of superpixels according to stroke width. The superpixel number K is set to

$$K = Kx \cdot Ky$$
$$Kx = W/ws$$
$$Ky = H/ws \quad (2)$$

where W and H are image width and height, respectively, and ws is the average stroke width estimated in the SW estimation step. Kx and K y are the number of superpixels in the horizontal and vertical directions. After that, s2 = N/K is the adaptive target superpixel size for an image. Adaptive compactness: The higher the compactness is, the more regular the superpixels will be. However, the high regularity means being insensitive to colour change and being more likely to generate superpixels across multiple textures. Many text images contain strokes with variant width. The high regularity may generate the superpixels across the background and thin strokes, which will cause the broken strokes. This is a specific problem of text images, and a lower compactness is better for this case. On the contrary, a text image with more consistent stroke width needs regular super pixels and can accept high compactness. Therefore, an adaptive compactness m is calculated based on the measurement of SW variation

$$m = 2.0 \cdot m_0 \ / \ 1 + e^{\gamma} \cdot (\tau - \tau 0) \quad (3)$$
$$\tau = \sigma/w_s \quad (4)$$

where m_0 is the reference compactness, which is set to 20. σ is the SW standard deviation in an image. To measure the SW variation of an image, the normalized SW standard deviation τ is calculated as (4). Here, τ0 is the mean of normalized SW standard deviation in an image set, which usually varied within the range of 0.2–0.6 and is set to 0.4 in this paper. Here, γ controls the scale range, which is set to 3.0. The section of experiment will describe how to choose m0 and γ.

c.1.2 Initial binarisation: The initial binarisation produces a binary text image that can also be regarded as coarse text segmentation. This step has two purposes: to provide a binary image for SW estimation and to support the final stroke super pixel verification. In initial binarisation, even global binarisation methods such as Otsu's work well in most cases. However, the better the binarisation result, the more effective its support to later steps. In this paper, we utilize a simple and effective binarisation method based on normalized stroke width described in [21]. **c.1.3 SW estimation:** Each text image contains many strokes, and SW estimation is used to give average stroke width ws and SW standard deviation σ. To improve SW estimation precision and efficiency, we utilized a run-length-based SW estimation method. First, we construct two run-length sets by scanning the binary image in the horizontal and vertical directions. Ri H and RV j represent the ith and jth run-length in two sets, respectively. Then, the SW map is generated as

W x, y = min

P x, y   Ri H, P x, y   RV j

(Ri H, RV j)   (5)

where if P(x, y) is a stroke pixel belonging to two run-lengths from different directions, the length of short run-length is closer to the real stroke width at this pixel. Finally, after assuming stroke width W(x, y) is subject to Gaussian distribution N ws, σ2 , we can estimate the parameters of mean and standard deviation.

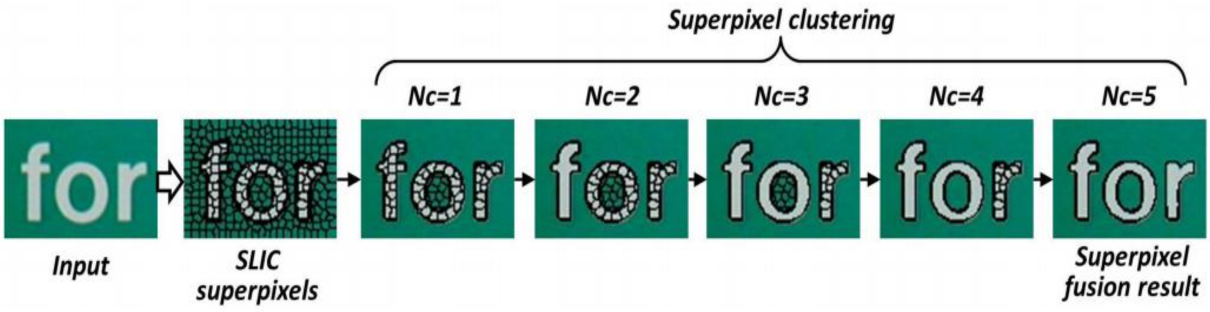
For the original image in Fig. 3, the estimated stroke width is 6.5 and the target K is 360. Therefore, its final superpixel image is similar to the third sub-figure which has K = 300.

C.2 Super pixel fusion

Larger image regions are more robust to noise and local disturbances. This is the foundation of effectiveness of superpixelbased representation. The purpose of superpixel fusion is to merge a large number of homogeneous superpixels to fewer, larger regions.

Super pixel clustering is applied to implement that. As previously mentioned, the ideal result is that every isolated stroke is entirely contained in one superpixel region and similar background texture components are also included in the same superpixel region. Those strokes and background textures have arbitrary shapes, and many superpixels belonging to same textures are much different in color space. From this point of view, the superpixel set of a text image is a kind of noisy non-Gaussian distribution data. Moreover, local disturbances and mutual interference exist between strokes and background textures. The fusion procedure is similar to clustering noisy non-Gaussian data and requires a robust clustering algorithm. DBSCAN [24], based on density analysis, can discover clusters of any data distribution without supervision and is highly robust to noisy data. These merits allow us to adopt and modify DBSCAN in this stage, and propose amodified-DBSCAN-based superpixel fusion method.

```
Input: Superpixel set SP and its size Nsp;
Output: Final superpixel set newSP;
Let N_c be the cluster number and be initialized to 0;
For i := 1 to Nsp Do
    If SP(i) has not been visited Then
        Mark SP(i) as visited;
        N_c = N_c+1;
        Assign SP(i) to the cluster N_c;
        Create the neighbor list NeighborList_1 for SP(i);
        j = 0;
        While j < the length of NeighborList_1 Do
            Let SP(k) be the j^th superpixel in NeighborList_1;
            If SP(k) has not been visited Then
                Mark SP(k) as visited;
                Create the neighbor list NeighborList_2 for SP(k);
                Append non-repetitive members from NeighborList_2 to the tail of NeighborList_1;
            Endif
            Assign SP(k) to the cluster N_c;
            j = j+1;
        End
    Endif
Endfor
Merge superpixels assigned to the same clusters into the new superpixels and construct the new superpixel set newSP;
```

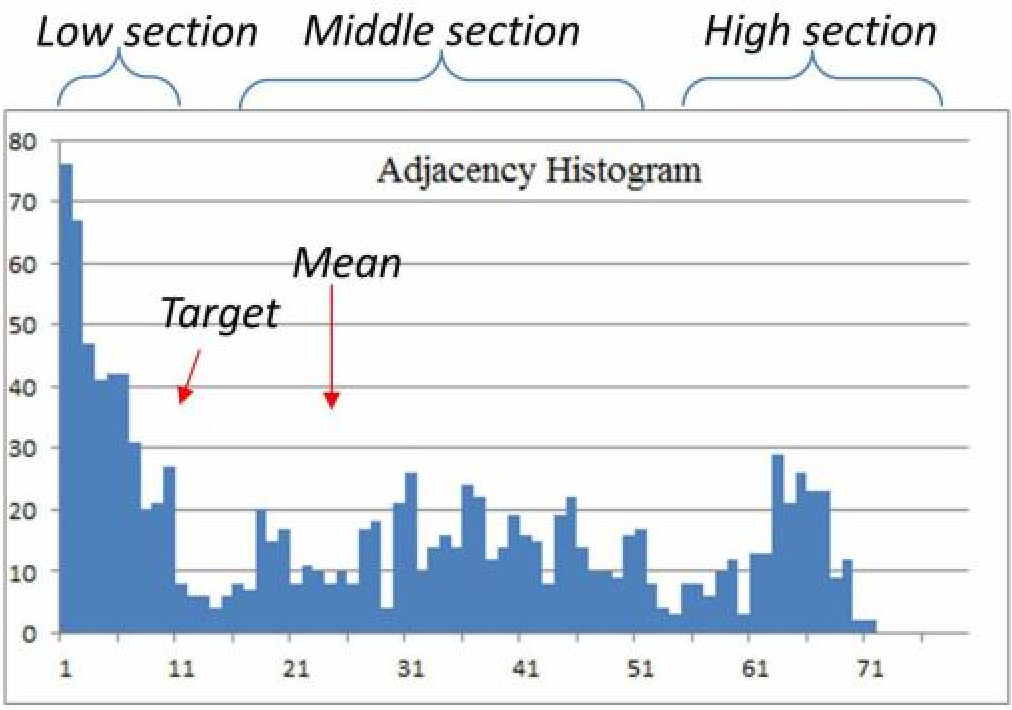


3.2.1 Modified-DBSCAN-based superpixel fusion: Starting with randomly picked data points, DBSCAN can grow regions with sufficiently high density into clusters of arbitrary shape. Its standard algorithm requires the input of two parameters Epsilon and MinPts. The neighborhood within a radius Epsilon of a given point is called the ε-neighborhood of the point and a point with at least MinPts of points within its ε-neighborhood is called a core point. The key concept behind DBSCAN-based superpixel fusion is to find density-reachable or density-connected superpixels and cluster them together into new superpixel regions. In contrast to regular SLIC superpixels, the new superpixels will be irregular in both of shape and size. Any superpixel can be a seed in clustering. A superpixel is represented by a 5D feature vector [L, a, b, x, y]T that includes two kinds of properties: position and colour. Since superpixels have various sizes and homogenous superpixels mainly depend on colour similarity, these two parts should be handled independently in distance measurement. Thus, we make two main modifications to the algorithm: Modification 1: the new criterion of neighborhood determination is: if two superpixels are adjacent in an image and their color distance is smaller than a threshold E c, they are neighbors. This criterion can also be expressed as (6)

$$N_{E_c}(p) = \{q \mid q \quad Adjacency\ p \cap d_c(p, q) \leq E_c\} \quad (6)$$

where Adjacency(p) is the adjacent superpixel set of superpixel p, which means that only adjacent superpixels are considered in neighbour searches. In this way, one round of density-reachable superpixel search can only merge adjacent superpixels and greater fusion depends on iterative cluster expansion. The neighborhood radius in colour space is Ec, which controls the adjacent superpixels that may be clustered together and is generally a Euclidean distance in the range 5–10. Modification 2: the density threshold MinPts is set to 1 to make all superpixels core points for clustering. Therefore, all neighbors are mutually density-reachable and an isolated superpixel can form a cluster even without neighbors. Moreover, the search can begin with any superpixel to cluster density-reachable superpixels. The superpixel fusion Algorithm 1 (see Fig. 4) is described as follows. Taking a simple image as an example, Fig. 5 illustrates superpixel fusion using this algorithm. Nc = 1 to 5 represents all SLIC superpixels being merged into five clusters. 3.2.2 Adaptive neighborhood radius Ec: The parameter of neighborhood radius Ec controls which superpixels are clustered together. Too small or large Ec will cause over-segmentation or under-segmentation in the final fusion result, respectively. Fig. 6 shows an adjacency histogram of color difference which usually has a three-peak distribution and can be roughly divided into three sections. The low section represents intra-homogenous-region differences. The high section represents differences from distinct superpixels. Most of values in the middle section are from differences between texture boundary superpixels and their adjacency. Thus, the proper Ec can separate the low section and middle section, and setting the adaptive Ec is a threshold search problem. The adjacency histogram H can be expressed as (7). To avoid the influence of the high section, a truncated adjacency

Adjacency histogram of colour difference



Histogram H2 is utilized as expressed in (8), where L is set to the

Mean of the histogram H. Here, p(i) in (9) are probabilities of each radius level in histogram H2. Then, the target Ec is the threshold maximizing the between-class variance δb 2 t between two classes separated by a threshold t as (10). δb 2 t is calculated as (11), where ω 0, 1 t and μ0, 1 t are probabilities and means of two classes

$$H(i) = \text{Count}(p, q \mid q \quad \text{Adjacency } p \cap dc\ p, q = i,\ 0 \leq I \leq 255 \quad (7)$$

$$H_2(i) = H(i),\ 0 \leq i < L \quad (8)$$

$$p\ i = H_2(i) / \sum_{j=0} H_{2(J)} \quad (9)$$

$$E_c = t\,\hat{} = \arg\max_{0 \leq t < L} \delta^2 b(t) \quad (10)$$

$$\delta^2 b(t) = \omega_0(t)\,\omega_1(t)[\mu_0(t) - \mu_1(t)]^2$$

$$\omega_0\ t = \sum_{i=0} p\ i$$

$$\omega_1\ t = \sum_{i=L} p\ i$$

$$\mu 0\ t = \sum_{i=0} I * p(i) / \omega_0$$

$$\mu 1\ t = \sum_{i=L} I * p(i) / \omega_1 \quad (11)$$

3.3 Stroke super pixel verification In stroke super pixel verification, every super pixel should be classified as text or background. Since all strokes and background textures are segmented accurately using superpixels, identifying them is not difficult. For a superpixel region, let NT and NB be the numbers of text and background pixels in the initial binary image, respectively. Therefore, to decide whether a superpixel belongs to a stroke, we can use the ratio of NT and NB or the rate of text pixels in the region. Let newCi represent the ith new super pixel after super pixel fusion. B(x, y) denotes the initial binary image and B(x, y) = 1 indicates a text pixel. The rate of text pixels in the superpixel area is calculated as

$$\varphi i = \sum_{P_j} \quad newCi\ B(x\,j, y\,j) / A_i \quad (12)$$

where if φi is larger than R, the corresponding superpixel new Ci is marked as a stroke region. Ai is the area of new Ci. The final text segmentation result is obtained by removing all background pixels and setting the stroke superpixels to black. Too large R will cause stroke missing while too small R may bring wrong strokes. Its impact is sample-dependent. A better result for one sample does not always lead to the same for other samples. That is, a balance is needed for all samples. Considering the effect of fine tuning on this parameter is slight, 0.5 is chosen for R. Fig. 7 shows intermediate results for two examples that resulted in clear and accurate final segmentation results. For the example on the right, stroke widths differ greatly between the horizontal and vertical strokes in a character, demonstrating that there is no fixed super pixel size parameter that can fit all local regions. Even in this case, the super pixel fusion is stable and ensures the good final result.

d. Region Based Segmentation Method: The region based segmentation techniques segment an image into several regions of similar features. This technique is based on the concept of uniformity. This approach is based on the fact that the neighboring pixels within a region have similar features. These features do not resemble the features of pixels of other regions. Generating a uniform region with larger size is the main objective of region based segmentation. This phenomenon generates very few regions within the image [6]. Although, the regions are similar in nature but some significant changes in the features of neighboring pixels can be noticed. Two basic methods based on this technique are described below:

i. Region growing methods: The region growing based segmentation methods segment an image into several regions on the basis of the growing of seeds (initial pixels). It is possible to select these seeds in manual way i.e. on the basis of earlier information or in automatic way i.e. on the basis of particular application. Afterward, the connectivity between pixels controls the seeds' growing. It can be stopped after getting the prior knowledge of issue.
ii. Region splitting and merging methods: The two basic techniques have been used by region splitting and merging based segmentation methods. These methods include splitting and merging for the segmentation of an image into several regions.Splitting iteratively divides an image into regions of similar features while merging combines the contiguous similar areas.

e. Clustering Based Segmentation Method: The clustering based methods segment an image into clusters that contain pixels of similar features. The process of dividing data elements into clusters so that elements within similar cluster are more similar to each other as compared to elements in other clusters is known as data clustering. There are two basic categories of clustering methods. These are Hierarchical method and Partition based method. The hierarchical methods depend on the concept of trees. The complete database is represented by the root of the tree in these methods. The clusters are represented by internal nodes. In contrast, the partition based methods iteratively use optimization methods for minimizing an objective function. Various algorithms are implemented between these methods for detecting clusters [7]. Clustering can be divided in two categories. These categories are:

i. Hard Clustering: It is a simple clustering method. This method partitions an image into set of clusters in such a way that one pixel belong to just one cluster. This means that every pixel may fit in just one cluster. Membership functions have been used by these methods. These functions contain either 1 or 0 value. These values provide information whether a certain pixel belongs to particular cluster or not. K-means clustering based algorithm called HCM is an example of a hard clustering based method. This approach initially computes all centers. After that, then every pixel is allotted to the nearby center. This approach maximizes the intra cluster similarity and minimizes the inter cluster equality.
ii. Soft clustering: The soft clustering is a more natural type of clustering. This is due to the fact that correct division cannot be done in real life because of the occurrence of noise. Therefore, soft clustering techniques are very valuable for segmenting those images in which division is not firm. Fuzzy c-means clustering is an example of this type approach. In this approach, the partitioning of pixels is done into clusters on the basis of partial membership. This means that one pixel may relate to more than one cluster. Membership values define the level of relatedness. This method is more flexible as compared to other methods.

f. Watershed Based Methods: The watershed based methods is based on the theory of topological interpretation. In this method, the basins having hole in their minima from where the water spills is represented by the intensity [8]. The neighboring basins are combined together after the reaching of water to the basin’s edge. The dams are needed for maintaining division between basins. These are the edges of the region of segmentation. These dams are built with the help of dilation. The watershed approaches use the gradient of image as topographic surface. The pixels with more gradient are signified as edges. These edges are continuous in nature.

g. Partial Differential Equation Based Segmentation Method: The partial differential equation based methods are considered as quick segmentation techniques. These methods are suitable for time based applications. Generally, two types of PDE techniques occur. These are non-linear isotropic diffusion filter and convex non-quadratic variation restoration. The non-linear isotropic diffusion filter is employed for edge enhancement while the other method is applied for noise removal. The PDE technique generates blurred boundaries. It is possible to shift these boundaries with the help of close operators. The fourth order PDE method is applied for reducing noise within image. The second order PDE method is utilized for detecting edges and boundaries in more efficient way.

h. Artificial Neural Network Based Segmentation Method: For decision making, the segmentation methods based on artificial neural network replicate the learning strategies of human brain. In recent times, this method is generally utilized for segmenting clinical images. This approach is employed for isolating the requisite image from background. A neural network consists of multiple connected nodes. Every connection has some specific weight. This approach does not depend on PDE. In this approach, the problem is converted to issues. Later, neural network is used to resolve these issues [9]. There are the two main steps included in this method. These steps include feature extraction and segmentation using neural network.

i. Model Based Segmentation:

i. Markov Random Fields Model: The Segmentation task depending on Markov Random Field (MRF) is known as Model based segmentation. MRF approach uses an inbuilt region smoothness limitation. This limitation is used for color segmentation. The element of the color pixel tuples are assumed as independent random variables for additional processing. The edge detection method is merged with MRF approach for accurate detection of edges. Markov Random Field (MRF) has limitation of spatial region evenness. Also, correlations occur between the color elements.

ii. Object-background Model: Object-Background models are based on histogram thresholding. They are unique models that carry out image segmentation. These models are based on the idea that a uniform background occurs and objects are positioned on this background in irregular manner. These models rely on shadowy features. Image histogram represents shadowy changes.

j. Fuzzy Theory Based Image Segmentation: Fuzzy set theory is utilized for image analysis. This approach provides accurate information from any image. The noise from image can be removed using Fuzzification function. A fuzzification function can easily converts a gray-scale image into a fuzzy image. In order to achieve more accurate results, several morphological operations can be merged with fuzzy approach. In image processing, Fuzzy k-Means and Fuzzy C-means (FCM) are the most commonly utilized approaches [10].

k. PDE Based Image Segmentation: In the field ofimage processing, PDE (Partial Differential Equations) equations or PDE are the most commonly used models. These models are specially used in image segmentation. These models make use of active contour model to carry out segmentation task. The segmentation issue is converted into PDE (Partial Differential Equations) using Active Contour model or Snakes.

Chapter 2

Theory and Methods

2.1 Super pixel Segmentation

Image representation is the prime step of image analysis. Superpixels represent the image into more logical way, categorize pixels on the basis of homogeneity criteria and restore edges. This phenomenon gives a different version of every image with irregular boundaries. The superpixels are different in sizes. Superpixels are the outcomes of an image over segmentation. Superpixels can be established as intermediate level image representation. There are various applications that use Superpixels. These applications include object detection, three dimensional reconstruction and semantic segmentation. A super pixel may be described as a spatially coherent homogeneous configuration [11].

The covering of an image is done with overlapping square patches of set size. Multiple patches cover every pixel. Assigning a pixel to these patches is a one among various tasks. No penalty is imposed in such a case when two neighboring pixels are allotted to the similar patch. If the pixels relate to different patches, in such a situation, stitching penalty is imposed. This penalty is inversely proportional to the intensity difference among the pixels. Instinctively, the stitching of patches is done for encouraging seems to be aligned with intensity edges. The stitching energy function regularizes the boundaries. A superpixel cannot be bigger than a patch size. Small superpixels are dejected because of their expensiveness in energy stitching.

2.2 Clustering Algorithms

In image segmentation, the process used to identify groups of similar image primitive is called clustering. Representing an image clearly is the major aim of clustering. In clustering, things or images with similar features are classified and distinguished. Clustering is a method of classification. In order to assign pixels to given quantity of clusters, there are various algorithms that provide image segmentation based on clustering. These algorithms include k-means algorithm, fuzzy c-means algorithm, moving k means algorithm, fuzzy moving k-means algorithm, adaptive moving k-means algorithm, adaptive fuzzy moving k means algorithm, Enhanced moving k-means algorithm. The processing of data is generally essential prior to the implementation of a clustering algorithm on image data. The processing ensures that it fulfills certain theories involved by the algorithm. Some particular part of information is extracted from the image using clustering [12].

A. Hierarchical Clustering: The clustering of the given image is carried out in this approach. This approach rely on the theory that pixel relates to nearby pixels more closely as compared to the farther pixels. In this type of approach, the grouping of the pixels is done into cluster on the basis of their distances. The data is represented into a tree like configuration by the Hierarchical clustering. The root node represents the complete data set. On the other hand, leaf node represents the individual data points. The similarity between the pixel data points is represented by the intermediate nodes within a tree configuration. The formation of different clusters is carried out at different stages in configuration. Therefore, cluster structures at respective stages are cut down for getting predictable clustering. Several algorithms are suggested on the basis of technique used to compute distances in this approach. The major branch of this type of clustering is known as agglomerative hierarchical clustering. This approach initiates with solo components in different patterns and combines it till the reaching of the stopping measure.

B. Partitioned clustering: The pixels or data points are divided into number of partitions using Partitional Clustering approaches. These divisions are termed as clusters. These types of clustering algorithms arrange data into sole partition rather than characterizing this data into nested configuration such as hierarchical clustering. It is more advantageous to use Partitional clustering for big data set. In big dataset, representing data in a tree configuration is not an easy job. There exist different types of partitional clustering algorithms. These algorithms include square error clustering, mode seeking clustering, mixture resolving clustering and Graph theoretic clustering.

i. K-means Clustering: K-Means is an unsupervised clustering algorithm. On the basis of intrinsic distance between input data points, this clustering algorithm classifies these data points into manifold classes. K means algorithm is one of the most commonly used iterative algorithms. This clustering algorithm is widely utilized due to its ease of use and convergence speed. In contrast to other clustering algorithms, this algorithm generates higher quality of clusters and requires less calculation. Minimizing the sum of squared distances among all points and the cluster center is the major objective of this clustering algorithm. It is a mathematical clustering algorithm. The technique generating groups of objects or clusters is known as data clustering. This clustering approach depends on the index of resemblance or difference amid pairs of data elements. It is an iterative, statistical, non-deterministic and unsupervised clustering algorithm. This clustering algorithm can be implemented on various data sets. However, this algorithm shows high quality performance with compact groups only. The particular amounts of disjoint, flat (non-hierarchical) clusters are generated by this algorithm. This clustering algorithm can generate circular clusters more efficiently.

This clustering algorithm classifies a particular data set using a certain number of clusters by fixing a priori in an easy and simple manner. Defining k centroids per cluster is the major thought here. As different locations generate different results, therefore, the placement of these centroids should be done cunningly. Hence, superior selection is to keep them as far away from each other as possible. In the next step, every point belongs to a given data set. This point is associated to the nearest centroid. When no point is pending, the first step is completed and an early group age is done. The first step is declared as completed after considering all points. At this moment, re-computation of k new centroids in form of bar centers of the clusters ensuing from the earlier step is essential [13]. It is necessary to perform a new binding between the same data set points and the nearest novel centroid after getting these k novel centroids. This results in the creation of a loop. Using this loop, it can be noticed that the k centroids move from their place after every step till no more movements occur. In different way, centroids do not change their place. At last, this algorithm focuses on minimizing an objective function. Here, this objective function is represented by a squared error function. There are three main drawbacks of this algorithm. These include:

- This algorithm does not determine K (the number of clusters).
- Different results are generated by different initial conditions.
- The data distant from center drag the centers away from optimal place.

C. Density-based Clustering: The cluster on the basis of the region's growing with high density is discovered by density based algorithms. These algorithms are called single-scan algorithms. On the whole, density-based approaches involve two algorithms. These are:

i. DBSCAN Algorithm: A density based clustering algorithm (DBSCAN) algorithm is based on the density-based idea of clusters. The density of points is considered for detecting clusters. The existence of clusters is represented by regions with a high density of points. On the other hand, regions with a low density of points represent noisy clusters or clusters of outliers. This algorithm is appropriate for large datasets with noise. This algorithm can recognize clusters of diverse magnitudes and shapes.

The DBSCAN algorithm is based on the concept that the neighborhood of a given radius should contain at least a minimal number of points for every point of a cluster. This implies that the density in the neighborhood should go beyond some predefined threshold level [14]. There are mainly three parameters required by this algorithm. These parameters include:

- Parameter K represent the size list of the neighbor
- Eps parameter represents radius that delimitate the neighbourhood region of a point (Epsneighbourhood).
- MinPts parameters represent the minimal number of points that should occur in the Eps-neighbourhood.

The clustering procedure classifies data points within the dataset as core points, border points and noise points. These data points are also classified as per the implementation of density associations between points (directly density-reachable, density-reachable, and density-linked) to generate the clusters.

Pixel-based evaluation

Pixel-based evaluation can give a very direct measure of the text

Segmentation performance and thus was chosen to evaluate the

KAIST dataset. Segmentation precision, recall, and F-measure

Were calculated and are defined as

Precision = $S \cap T / S$

Recall = $S \cap T / T$

$$F = 2 \times precision \times recall / precision + recall \quad (13)$$

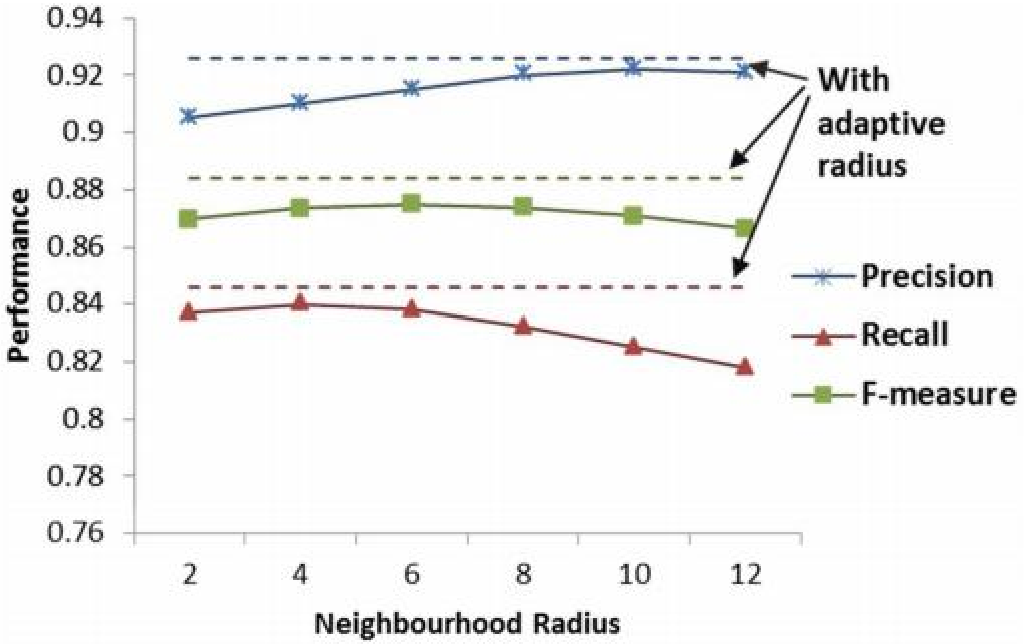
Where S represents the set of text pixels extracted from the test image, and T represents the set of text pixels in the ground-truth image. Fig. 9 depicts some of the text segmentation results of the proposed method. Super pixel fusion results are given in addition to the text segmentation results. In most cases, the superpixels after fusion contain an entire stroke or character, and the background textures are also clustered into as few super pixel regions as possible. This property is the basis of the method's robustness to noise and benefits non-stroke texture filtering as well. A comparison of several result images is shown in Fig. 10. The color polarity of text is automatically determined by Song et al. [28] for all images. Otsu's method discovers strokes in images with simple backgrounds, whereas some background textures are incorrectly labelled as text in complex or noisy images. This method was sensitive to variant lighting. The MSER performs well in many images, though they are unstable for some blurred images and complex background textures. The SWT method is apt at stroke-like textures discovering, but yields too much binary noises. Zhu's method performs poorly for text with wide strokes, and for images with strong lighting or complex background textures. The proposed method results in accurate strokes and a clear background even for complicated images. Using pixel-based evaluation, the performance of the different methods on cropped images from the KAIST database is shown in Table 1. The proposed method performs well in different sample categories. It achieved the best precision and F-measure in most categories, while MSER obtained the best recall in most categories. Generally, high precision indicates strong de-noising ability. The average precision was high, 0.926, while recall remained around 0.85, indicating that the proposed method is robust to noise and complex background textures. Observing the average F-measure of 0.884, the proposed method significantly outperforms the others more than 2.3 percentage points. Figs. 11a and b are bar graphs comparing the precision and recall of all methods and display the improvement in the results of the proposed method more intuitively. Although super pixel-based text segmentation's accuracy is very promising, its main disadvantage is its heavy computation costs. Compared to Otsu's method, which completes an image in an average of 1.4 ms, and MSER, which takes 28 ms, Zhu's method and the proposed method spent 329 and 393 ms on an image, respectively; the most time consuming step is SLIC super pixel generation. Owing to its significant improvement in text segmentation accuracy, this cost is acceptable for many applications and could be reduced greatly by using parallel computing in superpixel generation. Prior work has shown that SLIC superpixels can be computed in real time via a Graphic Processing Unit.

Recognition-based evaluation

To verify the generalization of the proposed method, experiments on two more datasets are carried out. Without text segmentation ground-truth, the ICDAR2003 and SVT datasets can only be evaluated using recognition results. The color polarity of text was automatically determined by [28] in advance too. To weaken the comparison, we followed the same settings with previous works [6, 7, 27]. First, we ignored all words that contain non-alphanumeric characters, as well as words with two or fewer characters. Then, the lexicons provided in [27] were adopted. For the ICDAR dataset, only the full lexicon with all the words in the test set was used in the experiment. For both two datasets, the proposed method outperformed other segmentation methods significantly. The learning-based text recognition methods got the highest word recognition accuracy. However, due to the good segmentation results of the proposed method, the off-the-shelf OCR engine achieved a close accuracy, 69.2%, on the dataset ICDAR2003. This result also indicates the importance of de-noising capability for character recognition. In contrast to the ICDAR2003 dataset having the similar image quality of the KAIST data, the SVT dataset contains many low-quality images, especially including lots of small text images. That is more challenging for text segmentation and resulted in the much lower recognition accuracy. It can also be observed that learning-based character features and specially trained recognition systems are effective to improve the final text recognition accuracy, especially for seriously degraded scene text

Table 1 Pixel-based evaluation on the KAIST dataset. The bold values denote the best precision, recall and F-measure values for each category

Image category	Otsu [9]			MSER [16]			SWT [13]			Zhu *et al.*[23]			Proposed method		
	P	R	F	P	R	F	P	R	F	P	R	F	P	R	F
A	0.612	0.683	0.646	0.737	0.785	0.760	0.834	**0.845**	0.839	0.922	0.792	0.852	**0.954**	0.826	**0.885**
B	0.608	0.657	0.632	0.691	**0.918**	0.788	0.759	0.819	0.787	**0.924**	0.839	0.879	0.910	0.853	**0.881**
C	0.834	0.813	0.823	0.789	0.820	0.804	0.787	0.791	0.789	0.905	**0.837**	0.870	**0.934**	0.830	**0.879**
D	0.748	0.753	0.750	0.788	**0.923**	0.850	0.870	0.892	0.881	0.888	0.838	0.862	**0.910**	0.821	**0.863**
E	0.580	0.625	0.602	0.885	**0.926**	**0.905**	0.885	0.783	0.831	0.900	0.781	0.836	**0.926**	0.808	0.863
F	0.865	0.855	0.860	0.775	0.882	0.825	0.883	**0.901**	0.892	0.917	0.851	0.883	**0.928**	0.865	**0.896**
G	0.774	0.828	0.800	0.725	0.764	0.744	0.812	0.785	0.798	0.879	0.849	0.864	**0.891**	**0.862**	**0.876**
H	0.556	0.780	0.649	0.628	**0.871**	0.730	0.608	0.732	0.664	0.727	0.794	0.759	**0.850**	0.818	**0.834**
I	0.780	0.785	0.783	0.830	0.834	0.832	0.890	**0.867**	0.878	0.899	0.825	0.860	**0.986**	0.855	**0.916**
J	0.936	0.891	0.913	0.792	0.909	0.847	0.909	**0.940**	**0.924**	0.942	0.873	0.906	**0.953**	0.880	0.915
K	0.943	**0.911**	**0.927**	0.738	0.894	0.809	0.894	0.909	0.902	0.927	0.873	0.899	**0.947**	0.883	0.914
average	0.749	0.780	0.762	0.762	**0.866**	0.808	0.830	0.842	0.835	0.894	0.832	0.861	**0.926**	0.846	**0.884**



Evaluation of adaptive parameters

The adaptive parameters play the important roles in the proposed method. They are super pixel size s2, compactness m, and neighbourhood radius Ec. To verify and measure their effectiveness, comparison experiments are carried out on the KAIST dataset. Table 3 gives the results of different super pixel size settings including four fixed values and the adaptive one. The adaptive size is better than all fixed ones. Table 4 compares the results of different fixed compactness. The best one, 20, is set to the reference compactness m0. Table 5 describes how to choose γ for the adaptive compactness in (3), where $\gamma = 3.0$ is selected. Table 6 and Fig. 12 illustrate the result comparison for different neighbourhood radius settings. It is observed that performances of fixed radiuses are lower than the adaptive one. In the end, Table 7 gives a summary of the effectiveness of above three adaptive parameters. After replacing the adaptive parameters by the best fixed parameters, losses of performances occur for each parameter. Compared to the full version with all adaptive parameters, the F measure differences of the proposed method without adaptive super pixel size, compactness, and neighbourhood radius are 0.008, 0.005, and 0.007, respectively. These experimental results verify that the adaptive parameters have contributed to the performance improvement for the proposed method.

Conclusions and discussion

This paper proposed a text segmentation method based on super pixel clustering. Using super pixel-based image representation, the proposed method utilizes super pixel clustering to improve text segmentation performance. First, an adaptive SLIC text super pixel generation method is proposed. With the adaptive parameters of super pixel size and compactness, it improves super pixel boundary adherence and can generate accurate super pixels for text images. Then, a modified-DBSCAN-based super pixel clustering is used to fuse stroke super pixels or background super pixels to form larger objects. The larger super pixels try to encompass the entirety of strokes or background textures, which brings the higher robustness. Finally, stroke super pixel verification is performed on the larger super pixel regions and the text segmentation result is obtained. Both the pixel based evaluation and recognition-based evaluation demonstrated that the proposed method significantly outperformed the state-of the-art method. The most important advantage of the proposed method is its robustness to noise and complex background textures. This is critical for camera-captured natural scene text image analysis in real applications. These promising experimental results demonstrate the effectiveness of the proposed method.

test.py test72.py ● copied_gui.py testing.py again_project.py main.py

D: > image-segmentation > main.py > ...

```
import argparse
import logging
import time
from graph import build_graph, segment_graph
from random import random
from PIL import Image, ImageFilter
from skimage import io
import numpy as np


def diff(img, x1, y1, x2, y2):
    _out = np.sum((img[x1, y1] - img[x2, y2]) ** 2)
    return np.sqrt(_out)


def threshold(size, const):
    return (const * 1.0 / size)


def generate_image(forest, width, height):
    random_color = lambda: (int(random()*255), int(random()*255), int(random()*255))
    colors = [random_color() for i in range(width*height)]

    img = Image.new('RGB', (width, height))
    im = img.load()
    for y in range(height):
        for x in range(width):
```

```
Terminal   Help                    • main.py - Visual Studio Code
test.py   test72.py ●   copied_gui.py   testing.py   again_project.py   main.py ●
D: > image-segmentation > main.py > ...
    img = Image.new('RGB', (width, height))
    im = img.load()
    for y in range(height):
        for x in range(width):
            comp = forest.find(y * width + x)
            im[x, y] = colors[comp]

    return img.transpose(Image.ROTATE_270).transpose(Image.FLIP_LEFT_RIGHT)


def get_segmented_image(sigma, neighbor, K, min_comp_size, input_file, output_file):
    if neighbor != 4 and neighbor!= 8:
        logger.warn('Invalid neighborhood choosed. The acceptable values are 4 or 8.')
        logger.warn('Segmenting with 4-neighborhood...')
    start_time = time.time()
    image_file = Image.open(input_file)

    size = image_file.size  # (width, height) in Pillow/PIL
    logger.info('Image info: {} | {} | {}'.format(image_file.format, size, image_file.mode))

    # Gaussian Filter
    smooth = image_file.filter(ImageFilter.GaussianBlur(sigma))
    smooth = np.array(smooth)

    logger.info("Creating graph...")
    graph_edges = build_graph(smooth, size[1], size[0], diff, neighbor==8)
```

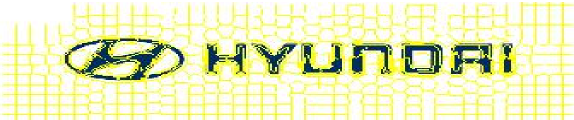
```
erminal   Help                    • main.py - Visual Studio Code
test.py   test72.py ●   copied_gui.py   testing.py   again_project.py   main.py ●   graph.py
D: > image-segmentation > main.py > ...
    logger.info("Creating graph...")
    graph_edges = build_graph(smooth, size[1], size[0], diff, neighbor==8)

    logger.info("Merging graph...")
    forest = segment_graph(graph_edges, size[0]*size[1], K, min_comp_size, threshold)

    logger.info("Visualizing segmentation and saving into: {}".format(output_file))
    image = generate_image(forest, size[1], size[0])
    image.save(output_file)

    logger.info('Number of components: {}'.format(forest.num_sets))
    logger.info('Total running time: {:0.4}s'.format(time.time() - start_time))


if __name__ == '__main__':
    # argument parser
    parser = argparse.ArgumentParser(description='Graph-based Segmentation')
    parser.add_argument('--sigma', type=float, default=1.0,
                        help='a float for the Gaussin Filter')
    parser.add_argument('--neighbor', type=int, default=8, choices=[4, 8],
                        help='choose the neighborhood format, 4 or 8')
    parser.add_argument('--K', type=float, default=10.0,
                        help='a constant to control the threshold function of the predicate')
    parser.add_argument('--min-comp-size', type=int, default=2000,
                        help='a constant to remove all the components with fewer number of pixels')
    parser.add_argument('--input-file', type=str, default="./assets/cover.jpg",
```

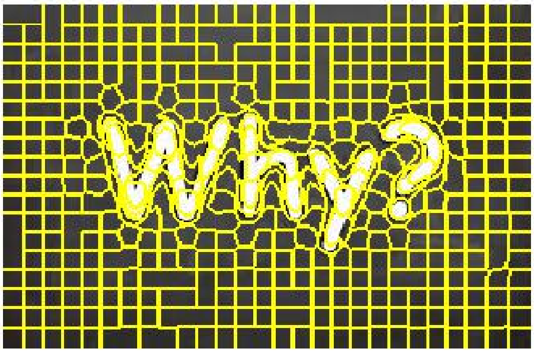
D: > image-segmentation > main.py > ...

```python
    parser.add_argument('--K', type=float, default=10.0,
                        help='a constant to control the threshold function of the predicate')
    parser.add_argument('--min-comp-size', type=int, default=2000,
                        help='a constant to remove all the components with fewer number of pixels')
    parser.add_argument('--input-file', type=str, default="./assets/cover.jpg",
                        help='the file path of the input image')
    parser.add_argument('--output-file', type=str, default="./assets/cover_out.jpg",
                        help='the file path of the output image')
    args = parser.parse_args()

    # basic logging settings
    logging.basicConfig(level=logging.INFO,
                    format='%(asctime)s %(name)-12s %(levelname)-8s %(message)s',
                    datefmt='%m-%d %H:%M')
    logger = logging.getLogger(__name__)

    get_segmented_image(args.sigma, args.neighbor, args.K, args.min_comp_size, args.input_file, args.output_file)
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL 1: python

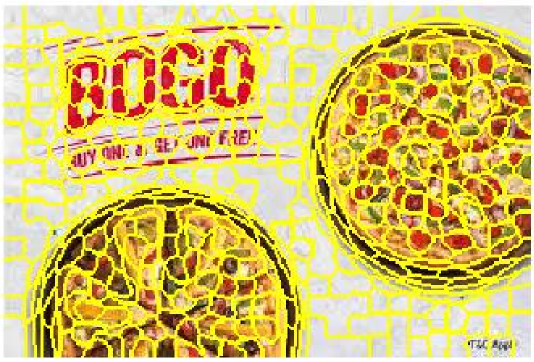
```
aaa.jpg         code.txt    image-segmentation                     pendrive

Rahul6@DESKTOP-2612K53 MINGW64 /d
$ cd image-segmentation

Rahul6@DESKTOP-2612K53 MINGW64 /d/image-segmentation
$ python main.py
01-09 09:44 __main__     INFO     Image info: JPEG | (2048, 831) | RGB
01-09 09:44 __main__     INFO     Creating graph...

Rahul6@DESKTOP-2612K53 MINGW64 /d
$ cd image-segmentation

Rahul6@DESKTOP-2612K53 MINGW64 /d/image-segmentation
$ python main.py
01-09 09:44 __main__     INFO     Image info: JPEG | (2048, 831) | RGB
01-09 09:44 __main__     INFO     Creating graph...
01-09 09:46 __main__     INFO     Merging graph...

$ python main.py
01-09 09:44 __main__     INFO     Image info: JPEG | (2048, 831) | RGB
01-09 09:44 __main__     INFO     Creating graph...
01-09 09:46 __main__     INFO     Merging graph...
01-09 09:46 __main__     INFO     Visualizing segmentation and saving into: ./assets/cover_out.jpg
01-09 09:46 __main__     INFO     Number of components: 84
01-09 09:46 __main__     INFO     Total running time: 149.7s
```

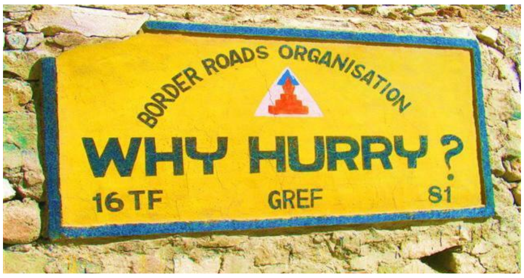

HYUNDAI

HYUNDAI


Why?

Why?


WHY SO
SERIOUS?

WHY SO
SERIOUS?


BOGO
BUY ONE & GET ONE FREE
*T&C Apply

BOGO



Chapter 3

Related Work

Youbao Tang, et.al (2018) proposed a new scene text detection technique. This technique involved super pixel-based stroke feature transform (SSFT) and deep learning based region classification (DLRC) [15]. In order to extract candidate character region (CCR), the development of SSFT (super pixel-based stroke feature transform) had been done. This approach used super pixel-based clustering for dividing an input image into various segments. This approach removed many regions based on predefined criterion assured via the characters. This approach computed a stroke width map for cleansing the residual regions to get the candidate character area. The joint training of deep feature extraction CNN and the feature fusion FCNs had been carried out in the in the DLRC step. Afterward, the extracted character regions were combined together for generating candidate text regions. These text regions were used for the detection of final scene texts. The three openly existing available dataset called ICDAR2011, ICDAR2013, and street view text were used in this work for the evaluation of recommended approach. F-measures of 0.876, 0.885, and 0.631 were achieved by the recommended approach correspondingly. These results demonstrated the efficiency of the recommended scene text detection approach.

Yuanping Zhu, et.al (2017) recommended a novel super pixel clustering based on text segmentation approach [16]. Initially, an adaptive simple linear iterative clustering-based text super pixel generation algorithm had been suggested for generating precise superpixels for text pictures. In order to improve edge accuracy, the computation of adaptive super pixel size and compactness had been carried out. Secondly, homogeneous superpixels were merged into bigger regions for both strokes as well as the background by the super pixel clustering. This was done to improve the overall exposure of strokes from superpixels. Also, this work presented an adapted density dependent spatial clustering applications with noise. At last, every region was allotted to a stroke or to the background after verifying stroke super pixel. This phenomenon provided the result of text segmentation. The recommended approach was quite robust to noise and complex background textures as per the achieved test results.

Soham Patel, et.al (2018) stated that a lot of computer vision applications used Super pixel. Super pixel was an image patch that fixed well to the image edges [17]. In contrast to rectangular patch, it showed better alignment with intensity boundaries. Superpixel had the ability to carry large number of information. Instinctively, superpixels partitioned the similar pixels for developing visually significant objects. It totally reduced the number of primitives to carry out subsequent image processing tasks. Image over-segmentation was the other name given to super pixel segmentation. Creating a reliable pixels grouping was the major purpose of super pixel segmentation. This main aim of this work was to give an overview of available cluster based super pixel segmentation methods. Moreover, these algorithms were compared in terms of different parameters for getting the direction for future work.

Inyong Yun, et.al (2019) recommended directional depth propagation for converting two dimensional images into three dimensional images [18]. For this purpose, color/depth-based super pixel segmentation approach had been used in this work. On the basis of motion information between two sequential frames, depth propagation generated the intensity image of the query edge. On the other hand, color-based super pixel segmentation caused several segmentation errors. This happened because of the occurrence of weak border and similar colors. The tested outcomes depicted that the recommended approach effectively estimated motion vectors for depth propagation. This approach performed better than state-of-the-arts techniques with analogous time cost in terms of PSNR (Peak Signal to Noise Ratio).

Giuseppe Masi, et.al (2015) recommended a novel object-oriented segmentation approach for high-resolution remote sensing pictures [19]. An appropriate watershed transform provided a preliminary super pixel depiction of the image for limiting computational complexity. Afterward, a region adjacency graph was connected with the superpixels using edge weights responsible for region resemblance/difference. A graph-cutting approach provided the last segmentation. Afterward, the formulation of a correlation clustering had been done. Several tests conducted on a realistic high-resolution remote sensing image proved the efficiency of the recommended technique.

Mohammed Q. Alkhatib, et.al (2018) recommended a super pixel-based hyper spectral image unmixing [20]. Initially, superpixel segmentation was implemented on the image. Every super pixel had been represented for developing a low dimensional representation of the image. Secondly, the segmentation of superpixel image had been done into regions. In order to retrieve local end members, End member extraction had been implemented for every region. The extracted end members had been used for measuring abundances over the complete image. The comparison of recommended technique had been done with the global unmixing and the local unmixing with the help of genuine picture. HYDICE Urban hyper spectral image was used in this work for providing the result of tests.

A. Abinaya, et.al (2016) designed a super pixel based segmentation algorithm for flower segmentation [21]. In contrast to pixel based approach, the recommended approach accurately provided better information. This approach improved the accuracy and reduced the difficulties in image processing. The superpixels were generated using Simple Linear

Iterative Clustering algorithm. DBSCAN clustering was used for the detection of flower cluster on the basis of super pixel image. This approach merged the clusters with nearby density. Jasmine images downloaded from the internet were used for the testing of segmentation of jasmine flowers.

Guoheng Huang, et.al (2017) proposed a video co-segmentation based super pixel co-saliency approach [22]. The recommended approach was based on a theory. The similar entities in numerous video clips were prominent, and they shared the analogous features. Initially, an attempt had been made for detecting the regions in all clips. A novel super pixel co-saliency approach had been recommended in this work. Afterward, multiple salient superpixels were selected as the original object marker superpixels. The tested outcomes depicted that the recommended model could successfully perform the segmentation of the common objects using multiple video clips. In contrast to several other existing approaches, the recommended approach showed low error rate.

Dawei Yang, et.al (2018) presented a cascaded super pixel pedestrian object segmentation algorithm [23]. This approach considered background interference. The secondary super pixel attained the association level of the standard color and center point. This phenomenon was based on Euclidean distance of every super pixel blocks amid inner and outer of the pedestrian saliency detection region for segmenting the upright person. The achieved simulation results demonstrated that the proposed algorithm showed precision-recall statistical average of 0.9797. In contrast to state-of-the-art saliency object segmentation algorithm, the recommended approach more efficiently achieved target extraction.

Xiaopeng Wang, et.al (2016) stated that brain tumor CT image segmentation generally suffered from the fuzzy boundaries [24]. Generally, the manual segmentation was based on the medical experience of the doctor. In this work, a novel technique for the segmentation of brain tumor CT image had been recommended. This approach was based on SLIC0 superpixels. The main purpose of recommended algorithm was to perform the segmentation of brain tumor in accurate manner. Initially, the superpixels were generated using simple linear iterative clustering with version 0 (SLIC0). After that, region merging was used for merging the analogous superpixels on the basis of their Gray. At last, the segmentation of the brain tumour regions had been carried out. The tested results depicted that the recommended approach could perform the segmentation of target tumour in accurate manner. After setting the pixel number of superpixels, it was possible to adjust the segmentation accuracy

Biao Hou, et.al (2018) recommended a decomposition-feature-iterative-clustering (DFIC) super pixel segmentation method for PolSAR pictures [25]. The recommended approach productively developed the decomposition features for creating superpixels. For this reason, the edges of super pixel segmentation were maintained efficiently. Two original PolSAR images were used for carrying out various tests. The tested outcomes demonstrated that the recommended approach performed better than several state-of-the-art super pixel techniques. In contrast to other pixel-based techniques, the DFIC super pixel-based classification approach achieved better outcomes.

Nannan Meng, et.al (2016) recommended a novel super pixel generation approach. The recommended approach was based on an error-controlled tactic [26]. Initially, the SLIC technique was utilized as a predefined step. The main aim of this step was to get a preliminary segmentation picture. Afterward, two thresholds were employed for carrying out region merging and mean pooling tasks in iterative manner. The tested results depicted that the recommended approach could accomplish a comparable exchange among the number of superpixels and image information depiction. The recommended approach could maintain the edge of the major object within the image along with the efficient superpixels' allocation.

Yiwei Tang, et.al (2019) aimed to modify the design of weights of Entropy Rate Super pixel segmentation (ERS) for which its three new versions were proposed [27]. On the basis of Euclidean Distance (ED), Spectral Angle Distance (SAD) and spectral Correlation Coefficient (COR), the three respective versions were designed. Further, the performances of these versions were analysed by designing two evaluation metrics. The conducted experiments and achieved results showed that finely segmented super pixels were obtained by all the proposed variants. In comparison to other previous versions, the proposed method provided better outputs.

Kuo-Chin Hung, et.al (2018) proposed a mechanism to identify the seams from the coarse to fine in iterative manner such that the undiscovered delicate seams could be identified [28]. The Sobel filters were used along with the edge energy map for constraining the seam boundary. Further, the optimal horizontal and vertical seams were identified iteratively. The dynamic programming technique was used to find the optimal seams. Further, the undiscovered seams were identified by the proposed method by checking the color homogeneity and the minimum width of super pixel within the iteration step. In comparison to other algorithms the proposed algorithm provided improvement in accuracy.

Huiping Lin, et.al (2018) proposed a new super pixel approach which was based on a new distance function [29]. Further, for the PolSAR images, this research aimed to design a super pixel seed updating mechanism. Expected numbers of superpixels were used to initialize the super pixel seeds. Further, on the basis of the distance function, the pixels were clustered iteratively. Then, on the basis of updating strategy, the super pixel seeds were updated. Based on the conducted experiments and achieved simulation results it was seen that the proposed approach provided an improvement in the exchange among boundary adherence and compactness in comparison to previous approaches.

Gözde Nur Yeşilyurt, et.al (2017) presented a study related to the super pixel segmentation approach known as hyper spectral segmentation [30]. Here, the hyper spectral images are adapted with the simple linear iterative clustering based super pixel segmentation approach. This research also evaluated the effects of different distance metrics on the proposed scenario. The PCA, Euclidian distance, spectral divergence and spectral angular distance were few of the metrics on which experiment was conducted. The outcomes showed improvement in the overall result outcomes as compared to the existing methods.

Arun M. Saranathan, et.al (2016) proposed a novel graph-based agglomerative mechanism in which a threshold was set for the highest variability within the segments by enforcing segment uniformity [31]. A statistical analysis of within-class and in between in the class spectral divergences was used to calculate the threshold value. The parsimonious segmentations were generated and the calculation of accurate mineralogical summaries was facilitated by conducting the experiments using proposed algorithm. Highly effective outcomes were achieved through this experiment.

Chapter 4

Plan of Work

4.1 Problem Formulation

This research work is based on the image segmentation. The image segmentation methods are broadly classified into threshold based segmentation and region based segmentation. The k-mean segmentation is the region based segmentation algorithm which can define the number of centroids and cluster the data based on the Euclidean distance. The technique which is designed in the base paper use set of features for the segmentation which is called candidate character region set for the segmentation. The candidate character region set of use various features for the segmentation which are called mean, standard deviation, number of connected components etc. The extracted features can be fused together for the training and CNN can classify the features. To increase accuracy of segmentation, the region based and threshold based segmentation techniques can be merged together to design hybrid image segmentation techniques

4.2 Objectives

Following are the various objectives:-

1. To study and analyse various image segmentation techniques
2. Design hybrid segmentation technique for the image segmentation
3. Implement hybrid segmentation technique and compare with existing in terms of certain parameters

4.3 Research Methodology

This research is related to image segmentation and image segmentation techniques are broadly classified into region based and threshold based techniques. The region based segmentation techniques are those which can segment the image based on the textual features of the images. The images have various type of textural features like energy, entropy etc. The threshold based segmentation techniques are those which can segment the image into two segments. The pixels which have value above the threshold will be segmented into one segment and other into the second. In this research work the threshold based segmentation and region based segmentation will be merged together to form hybrid image segmentation technique. In the proposed methodology the threshold based segmentation techniques will be applied which can form two segments of the original image. The one segment will be proposed further and region based segmentation will be applied which can segment image into N number of regions based on the pixel similarity.

When the image gets segmented the CNN method will be applied for the classification of the region.

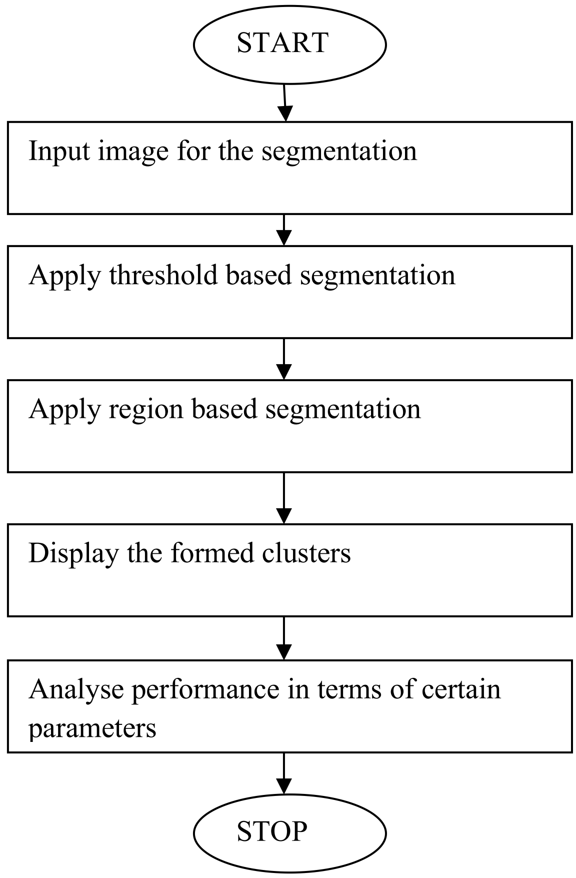


Figure 4.1: Proposed Flowchart

Bibliography

[1] A. P. James and B. V. Dasarathy,Medical image fusion: A survey of the state of the art”, Information Fusion, vol. 19, no. Supplement C, pp. 4-19, 2014.

[2] M. Vallières, C. R. Freeman, S. R. Skamene, and I. E. Naqa, “A radiomics model from joint FDG-PET and MRI texture features for the prediction of lung metastases in soft-tissue sarcomas of the extremities”, Physics in Medicine & Biology, vol. 60, no. 14, p. 5471, 2015.

[3] O. Li, and V. D. Calhoun, “Multi-set canonical correlation analysis for the fusion of concurrent single trial ERP and functional MRI”, NeuroImage, vol. 50, no. 4, pp. 1438-1445, 2010.

[4] X. Xu, D. Shan, G. Wang, and X. Jiang, “Multimodal medical image fusion using PCNN optimized by the QPSO algorithm”, Appl. Soft Comput., vol. 46, no. C, pp. 588-595, 2016.

[5] G. Bhatnagar, Q. M. J. Wu, and Z. Liu, “Directive Contrast Based Multimodal Medical Image Fusion in NSCT Domain”, IEEE Transactions on Multimedia, vol. 15, no. 5, pp. 1014- 1024, 2013.

[6] R. Singh and A. Khare, “Fusion of multimodal medical images using Daubechies complex wavelet transform - A multi-resolution approach”, Information Fusion, vol. 19, no. Supplement C, pp. 49-60, 2014.

[7] Y. Yang, “Multimodal Medical Image Fusion through a New DWT Based Technique”, in 2010 4th International Conference on Bioinformatics and Biomedical Engineering, 2010, pp. 1-4.

[8] X. Zhu, H.-I. Suk, S. W. Lee, D. Shen and I., “Subspace Regularized Sparse MultiTask Learning for Multi-Class Neurodegenerative Disease Identification”, IEEE transactions on bio-medical engineering, vol. 63, no. 3, pp. 607-618, 2016.

[9] S. Klein, U. A. van der Heide, I. M. Lips, M. van Vulpen, M. Staring, and J. P. W. Pluim, “Automatic segmentation of the prostate in 3D MR images by atlas matching using localized mutual information”, Medical Physics, vol. 35, no. 4, pp. 1407-1417, 2008.

[10] H. Cai, R. Verma, Y. Ou, S. k. Lee, E. R. Melhem, and C. Davatzikos, "Probalistic Segmentation of Brain Tumors based on Multi-Modality Magnetic Resonance Images", in 2007 4th IEEE International Symposium on Biomedical Imaging: From Nano to Macro, pp. 600-603, 2007

[11] SumitKaur, Dr. R.K Bansal, "Comparative Analysis of Superpixel Segmentation Method", 2018, International Journal of Engineering Technologies and Management Research

[12] Tatiraju, Suman, and Avi Mehta, "Image Segmentation using kmeans clustering, EM and Normalized Cuts", University Of California–Irvine, 2008

[13] Dalton, Lori, Virginia Ballarin, and Marcel Brun, "Clustering algorithms: on learning, validation, performance, and applications to genomics", Current genomics 10.6 (2009): 430.

[14] Deng Y. and Manjunath B.S., "Unsupervised segmentation of color-texture regions in images and video", 2001, IEEE Trans. Pattern Analysis Machine Intell, 23(8): 800-810.

[15] Youbao Tang, Xiangqian Wu, "Scene Text Detection Using Superpixel-Based Stroke Feature Transform and Deep Learning Based Region Classification", 2018, IEEE Transactions on Multimedia, Volume: 20, Issue: 9

[16] Yuanping Zhu, Kuang Zhang, "Text segmentation using superpixel clustering", 2017, The Institution of Engineering and Technology

[17] Soham Patel, Bintu Kadhiwala, "Comparative Analysis of Cluster Based Superpixel Segmentation Techniques", 2018, 2nd International Conference on Trends in Electronics and Informatics (ICOEI)

[18] Inyong Yun, Byunghyun Kwon, Joongkyu Kim, Cheolkon Jung, "Bi-Directional Depth Propagation for 2D-to-3D Conversion with Color/Depth-Based Superpixel Segmentation", 2019, International Conference on Electronics, Information, and Communication (ICEIC)

[19] Giuseppe Masi, Raffaele Gaetano, Giovanni Poggi, Giuseppe Scarpa, "Superpixel-based segmentation of remote sensing images through correlation clustering", 2015, IEEE International Geoscience and Remote Sensing Symposium (IGARSS)

[20] Mohammed Q. Alkhatib, Miguel Velez-Reyes, "Superpixel-Based Hyperspectral Unmixing with Regional Segmentation", IGARSS 2018, IEEE International Geoscience and Remote Sensing Symposium

[21] A. Abinaya, S. Mohamed Mansoor Roomi, "Jasmine flower segmentation: A superpixel based approach", 2016, International Conference on Communication and Electronics Systems (ICCES)

[22] Guoheng Huang, Chi-Man Pun, "Robust Video Co-Segmentation Based on Co-Saliency of Superpixels", 2017, 14th International Conference on Computer Graphics, Imaging and Visualization

[23] Dawei Yang, Junda Huang, Jing Zhang, Rubo Zhang, "Cascaded superpixel pedestrian object segmentation algorithm", 2018, Chinese Control And Decision Conference (CCDC)

[24] Xiaopeng Wang, Peng Ma, JunJun Zhao, "Brain tumor CT image segmentation based on SLIC0 superpixels", 2016, 9th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)

[25] Biao Hou, Chen Yang, Bo Ren, Licheng Jiao, "Decomposition-Feature-Iterative-Clustering-Based Superpixel Segmentation for PolSAR Image Classification", 2018, IEEE Geoscience and Remote Sensing Letters, Volume: 15, Issue: 8

[26] Nannan Meng, Fang Meng, Meng Chen, "An error-controlled method for superpixel segmentation", 2016, 9th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)

[27] Yiwei Tang, Liaoying Zhao, Lang Ren, "Different Versions of Entropy Rate Superpixel Segmentation for Hyperspectral Image", 2019, IEEE 4th International Conference on Signal and Image Processing (ICSIP)

[28] Kuo-Chin Hung, Chang-Hong Lin, “Iterative Superpixel Segmentation Based on Color Differences”, 2018, IEEE 7th Global Conference on Consumer Electronics (GCCE)

[29] Huiping Lin, Junliang Bao, Junjun Yin, Jian Yang, “Superpixel Segmentation with Boundary Constraints for Polarimetric SAR Images”, IGARSS, 2018 IEEE International Geoscience and Remote Sensing Symposium

[30] Gözde Nur Yeşilyurt, Alp Ertürk, Sarp Ertürk, “Metric and transform performance analysis for hyperspectral superpixel segmentation”, 2017, 25th Signal Processing and Communications Applications Conference (SIU)

[31] Arun M. Saranathan, Mario Parente, “Uniformity-Based Superpixel Segmentation of Hyperspectral Images”, 2016, IEEE Transactions on Geoscience and Remote Sensing, Volume: 54, Issue: 3