Spring 2017

Nuclei segmentation of histology images based on deep learning and color quantization and analysis of real world pill images

Sudhir Sornapudi

Department:

NUCLEI SEGMENTATION OF HISTOLOGY IMAGES BASED ON DEEP LEARNING AND COLOR QUANTIZATION AND ANALYSIS OF REAL WORLD PILL IMAGES

by

SUDHIR SORNAPUDI

A THESIS

Presented to the Faculty of the Graduate School of the

MISSOURI UNIVERSITY OF SCIENCE AND TECHNOLOGY

In Partial Fulfillment of the Requirements for the Degree

MASTER OF SCIENCE IN COMPUTER ENGINEERING

2017

Approved by

R. Joe Stanley, Advisor
Randy H. Moss
William V. Stoecker

ABSTRACT

Medical image analysis has paved a way for research in the field of medical and biological image analysis through the applications of image processing. This study has special emphasis on nuclei segmentation from digitized histology images and pill segmentation. Cervical cancer is one of the most common malignant cancers affecting women. This can be cured if detected early. Histology image feature analysis is required to classify the squamous epithelium into Normal, CIN1, CIN2 and CIN3 grades of cervical intraepithelial neoplasia (CIN). The nuclei in the epithelium region provide the majority of information regarding the severity of the cancer. Segmentation of nuclei is therefore crucial. This paper provides two methods for nuclei segmentation. The first approach is clustering approach by quantization of the color content in the histology images uses k-means++ clustering. The second approach is deep-learning based nuclei segmentation method works by gathering localized information through the generation of superpixels and training convolutional neural network.

The other part of the study covers segmentation of consumer-quality pill images. Misidentified and unidentified pills constitute a safety hazard for both patients and health professionals. An automatic pill identification technique is essential to address this challenge. This paper concentrates on segmenting the pill image, which is crucial step to identify a pill. A color image segmentation algorithm is proposed by generating superpixels using the Simple Linear Iterative Clustering (SLIC) algorithm and merging the superpixels by thresholding the region adjacency graphs. The algorithm manages to supersede the challenges due to various backgrounds and lighting conditions of consumer-quality pill images.

ACKNOWLEDGMENTS

Firstly, I would like to express my sincere appreciation to my advisor, Dr. R. Joe Stanley, for the constant guidance, encouragement, and support he has provided throughout my masters' study. He always showed me the right path to successfully complete my research and I am truly grateful for his support. I am also grateful to my committee members Dr. Randy H. Moss and Dr. William V. Stoecker for their continuous guidance and support.

Secondly, I would like to thank all my friends and colleagues at Missouri University of Science and Technology, for their encouragement and moral support by sharing their ideas that made me work on various approaches to a problem, which made my research much more enjoyable. I would like to thank Haider A., Jason H., Peng G., and Ravali E. for their inspiring thoughts and company during my work.

Finally, I am grateful to my parents, whose love and affection is always a blessing. It was their encouragement and support that constantly gave me courage to lead my life.

This research was supported by the National Library of Medicine (NLM).

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

Page

LIST OF TABLES

Page

NOMENCLATURE

Symbol	Description
σ	Standard Deviation
σ^2	Variance
μ	Accuracy

1. INTRODUCTION

The reconstruction of medical images into a digital form has propelled the fields of medical and laboratory research and clinical practice [1]. Image processing for medical image applications, specifically nuclei segmentation from digitized histology images and pill segmentation, has numerous challenges to attain highly accurate segmentation results.

In recent years, there have been a number of cervical cancer cases reported all over the world. This is the second most common cancer in women [2]. There is a cure for cervical cancer if it is detected early. The standard diagnostic process is the microscopic evaluation of histology images by a qualified pathologist [3]. The severity of cervical cancer increases as the immature atypical cells in the epithelium region increase. Based on this observation, the cancer affecting squamous epithelium is classified as Normal or three grades of cervical intraepithelial neoplasia (CIN): CIN1, CIN2, and CIN3 [4]–[6]. This can be clearly observed from Figure 1.1. Normal means there is no CIN, CIN1 corresponds to mild dysplasia (abnormal change), CIN2 denotes moderate dysplasia, and CIN3 corresponds to severe dysplasia.

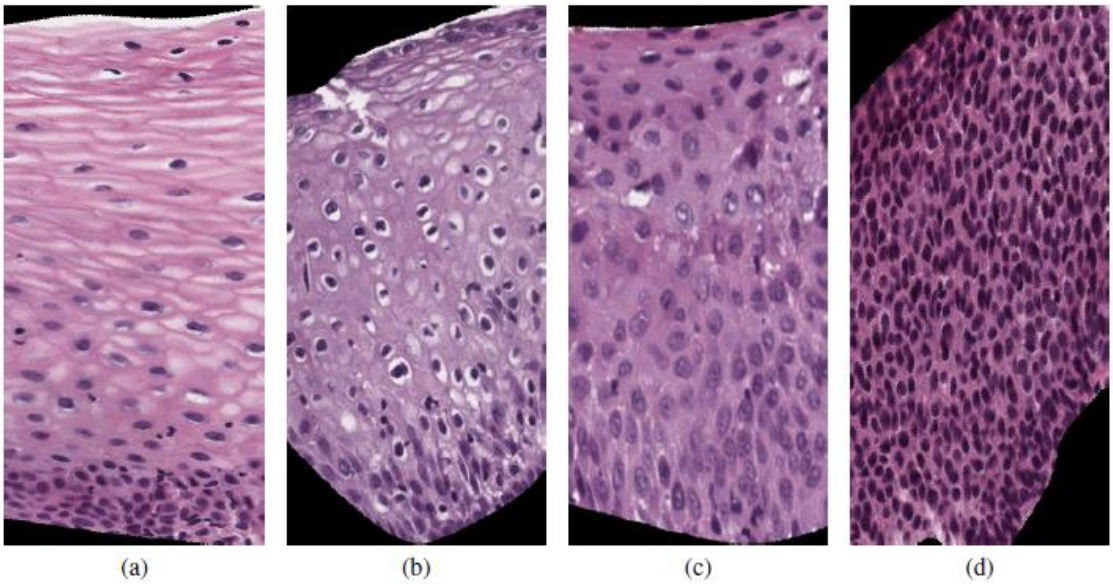

Figure 1.1 Samples of different CIN grades (a) Normal, (b) CIN1, (c) CIN 2, (d) CIN 3

As the severity of the cancer increases, an increase in the number of immature atypical cells can be observed from the bottom (basal layer) to the top of the epithelium region. This can be observed from Figure 1.1. Atypical immature cells are denser in the bottom region of the epithelium for CIN 1 [see Figure 1.1(b)]. For CIN 2, two-thirds of the bottom region is affected by the atypical immature cells [see Figure 1.1(c)]. Finally, for CIN 3, the atypical immature cells are densely spread over the whole epithelium region [see Figure 1.1(d)].

At present, cervical cancer tissue is analyzed manually. This is only done by pathologists with significant experience in their domain. These qualified pathologists are few in number, and it takes a considerable amount of time to observe the cancerous tissue. This calls for automatic histology image classification, which enables anyone with the images to easily verify whether the tissue is at a normal level or malignant.

Feature extraction plays a pivotal role in classifying images. The presence of nuclei in the epithelium allows the required data to generate various features in order to classify the cervical images. Hence, the generation of nuclei masks through proper nuclei segmentation of the cervical images is crucial. The paper employs two different methods to segment nuclei. The first method uses color quantization with the help of K-means++ clustering. The other method is based on deep learning through the extraction of superpixels and training a convolution neural network with the obtained superpixel data.

The paper also covers the challenge of real-world pill segmentation by superpixel merge. Misidentified or unidentified prescription pills are an increasing challenge for all caregivers, both families and professionals. Errors in pill identification may lead to serious or fatal adverse events. To respond to this challenge, a fast and reliable automated pill identification technique is needed. The first and most critical step in pill identification is segmentation of the pill from the background. The goals of segmentation are to eliminate both false detection of background area and false omission of pill area. Introduction of either type of error can cause errors in color or shape analysis and can lead to pill misidentification. The real-world consumer images used in this research provide significant segmentation challenges due to various backgrounds and lighting conditions. This paper proposes a color image segmentation algorithm which generates superpixels using the simple linear iterative clustering (SLIC) algorithm and merges the

superpixels by thresholding the region adjacency graphs. Post-processing steps are given to result in accurate pill segmentation. The segmentation accuracy is evaluated by comparing the consumer-quality pill image segmentation masks to the high quality reference pill image masks.

The remainder of this thesis is organized as follows. Section 2 presents a K-means++ clustering approach for nuclei segmentation in digitized histology images. Section 3 presents a deep learning approach for nuclei segmentation. Section 4 presents image processing techniques applied to pill segmentation. Section 5 presents study conclusions.

2. NUCLEI SEGMENTATION USING K-MEANS++ CLUSTERING

The proposed algorithm employs a color quantization approach for segmenting nuclei of histology images using a K-means++ clustering algorithm on the intensity of color pixels to reduce to four color intensity values. The flowchart in Figure 2.1 summarizes the procedure.

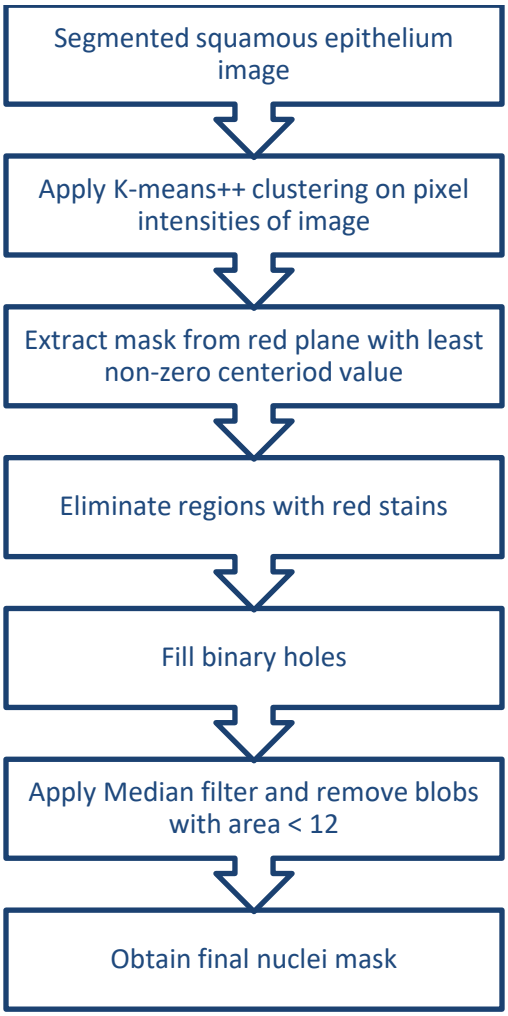

Figure 2.1 Overview of nuclei segmentation using color quantization with K-means++ clustering

The dataset images used in this paper are provided by the National Library of Medicine (NLM) as part of the research on classification of cervical histology images into Normal, CIN1, CIN2, and CIN3. A 71-image set is considered to segment the nuclei,

analyze the segmentation accuracy, and compare the results with previously published nuclei masks.

Color image quantization is the process of reducing a true color image into a number of distinct colors. This smaller set of colors represents the color properties of the original image. There are various approaches such as uniform quantization, median-cut, octree, clustering and popularity algorithm are used to obtain the color quantized image. According to the requirements of the problem, one approach will be chosen as the best choice.

There are four steps to the color image quantization process [7]:

1. Obtain color statistics by sampling the original image.
2. Chose a color map based on the obtained color statistics.
3. Use color map as a reference map for the pixel color values.
4. Draw the quantized image.

Color quantization can be divided into two classes: uniform and nonuniform. In uniform color quantization, the color space is partitioned to equal sized regions. This approach may be quick at a cost of the quality of the image. In non-uniform color quantization, the color space is adaptively partitioned based upon the distribution of colors.

Clustering is grouping data with similar properties, which is one of the classic problems in computational geometry and machine learning. K-means clustering is a way of choosing k centers from a given set of n data points to minimize the sum of squared distances φ between chosen centers and the data points. The K-means clustering algorithm is an unsupervised technique to group data into k clusters based on similarity in the data. K-means++ clustering is a modified form of the K-means algorithm in which seeds are initialized in a particular way such that it has better speed and accuracy compared to standard K-means.

The paper uses K-means clustering to group the color intensity values of pixels present in the image into k clusters. The algorithm randomly initializes the cluster centers and searches the entire image to find the near color intensity values within a residual error value. For each and every iteration, new cluster centroids are formed, which is the median of color values present in the respective clusters. This process takes place until N

iterations, and finally a labeled image is generated with each cluster having the color intensity value of its respective centroid color value.

Figure 2.2 shows a sample where a superpixel extracted from the histology image is processed with K-means color quantization to obtain 4 colors within 10 iterations.

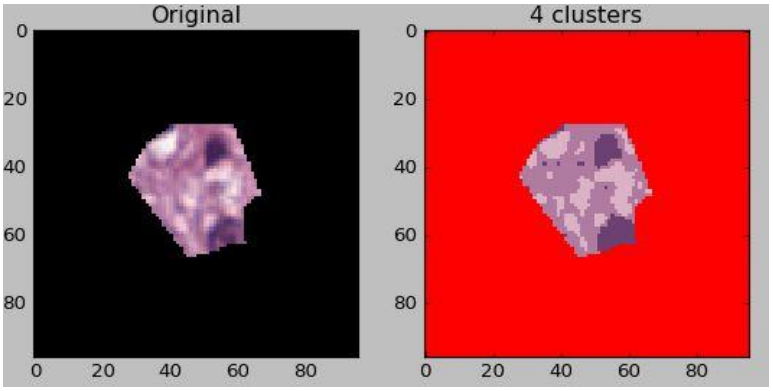

Figure 2.2 K-means++ color quantization on a superpixel

As mentioned earlier 71 images provided by NLM are used in the segmentation and analysis. The segmentation process is explained in detail by working on one of the images in the dataset. Figure 2.3 shows a sample cervical histology image and the manually segmented epithelium is shown in Figure 2.4.

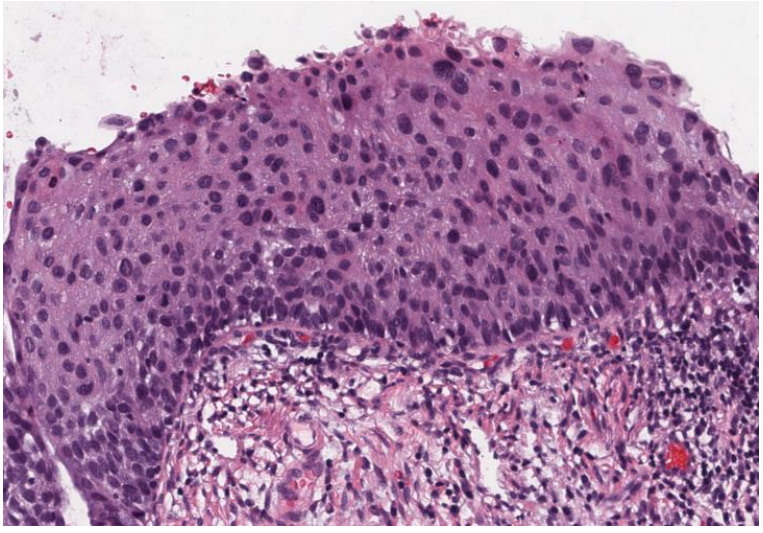

Figure 2.3 Original image sample

Figure 2.4 Epithelium segmented image

K-means++ clustering is applied in the RGB plane on color pixel data of the epithelium-segmented image. The parameters for clustering are selected such that the color intensities of the original image are mapped to 4 unique color values with 10 iterations. For the visualization of the clustered data shown in Figure 2.5, the pixel values of red plane are plotted against pixel values of grayscale image. The points in blue and red from the graph are the pixels of nuclei region. The pixel data is finally quantized such that the data points in each cluster are assigned the centroid value of the corresponding cluster.

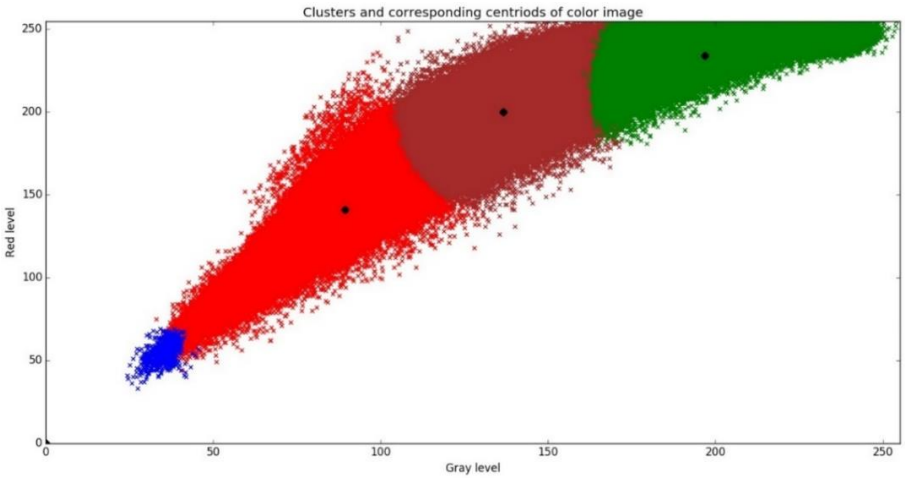

Figure 2.5 Graphical representation of cluster data

The nuclei mask is obtained by thresholding the data points in the red plane with the least nonzero centroid value, that is the centroid value from the cluster in red from Figure 2.5. The resultant mask is shown in Figure 2.6.

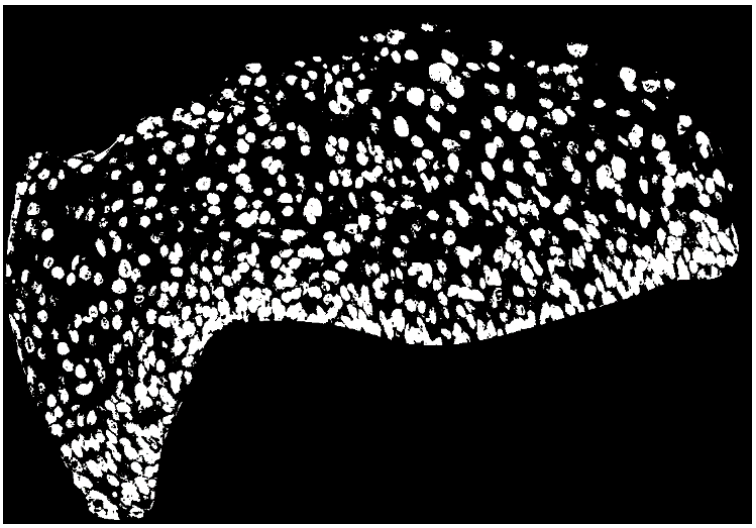

Figure 2.6 Nuclei mask as a result of thresholding the red plane

The red stains on some of the cervical tissue sample images are also detected as nuclei. The red regions are detected from the original image by assigning the range of upper and lower color boundaries. A mask is automatically created with the assigned boundary limits as shown in Figure 2.7.

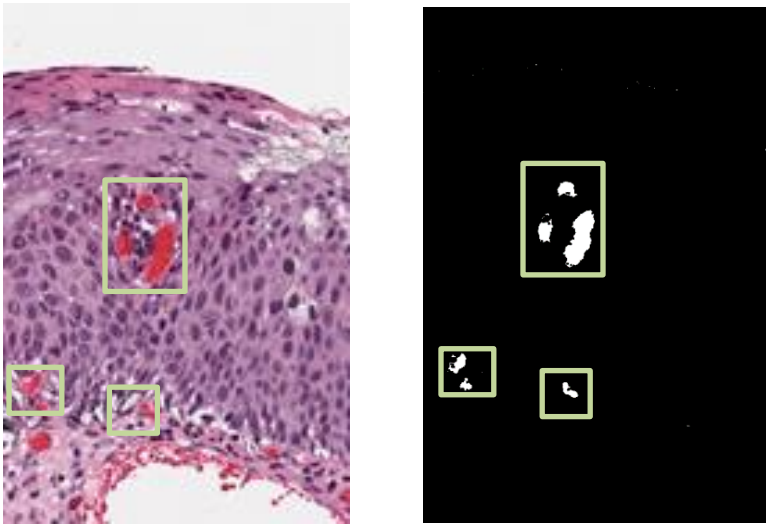

Figure 2.7 Red stains marked on image sample (left) and its respective mask (right)

The resultant red stain mask is complimented and AND operated with the obtained nuclei segmented binary image as shown in Figure 2.8 (left).

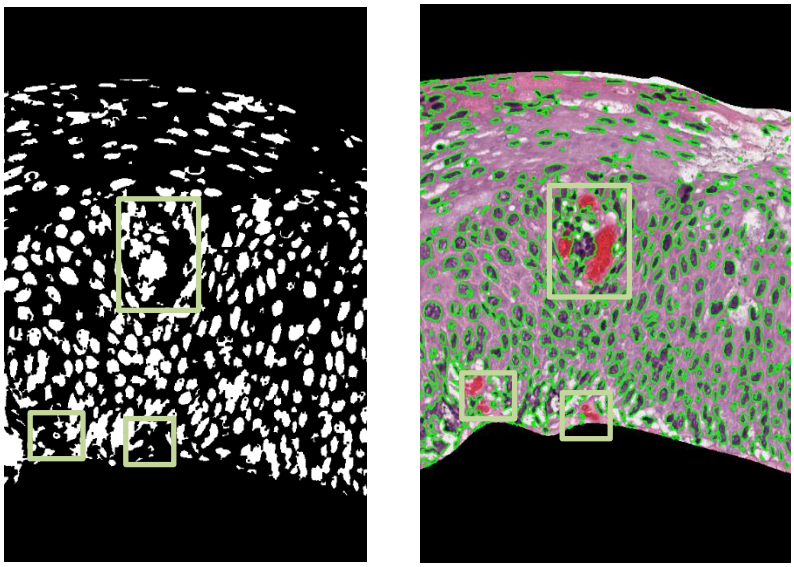

Figure 2.8 Nuclei mask without red stains (left) and segmented original image (right)

The obtained mask is then modified to fill the holes, if any, in the objects. A morphological closing operation is then applied to merge any irregular openings in the

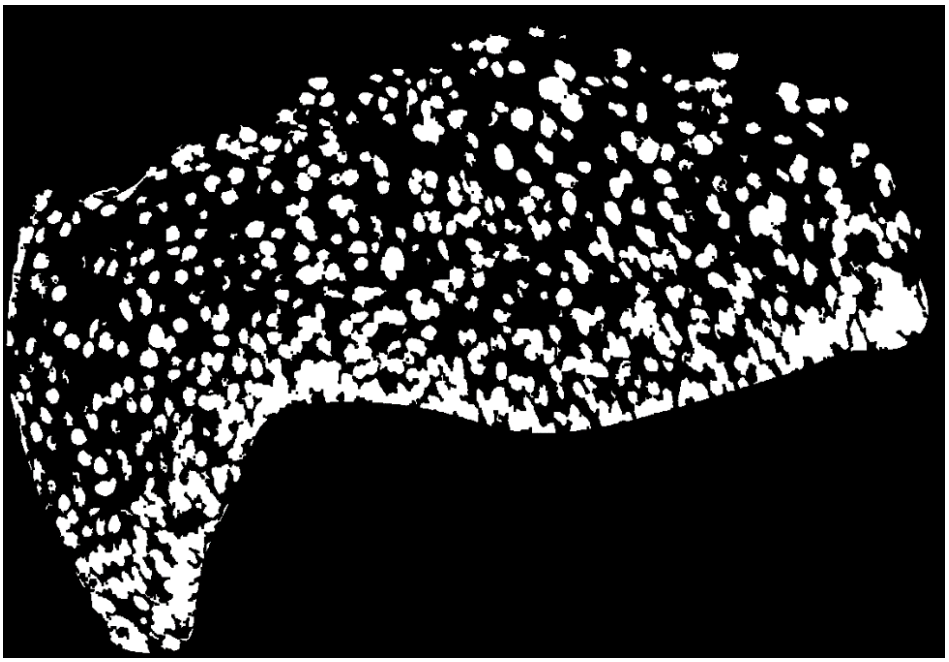

Figure 2.9 Final nuclei mask of cervical image

mask, objects with smaller area are eliminated by extracting the region properties, resulting in final nuclei mask as shown in Figure 2.9.

The obtained mask is used to detect the contours around the nuclei which are then superimposed on the original image using green color boundary for better visualization as shown in Figure 2.10.

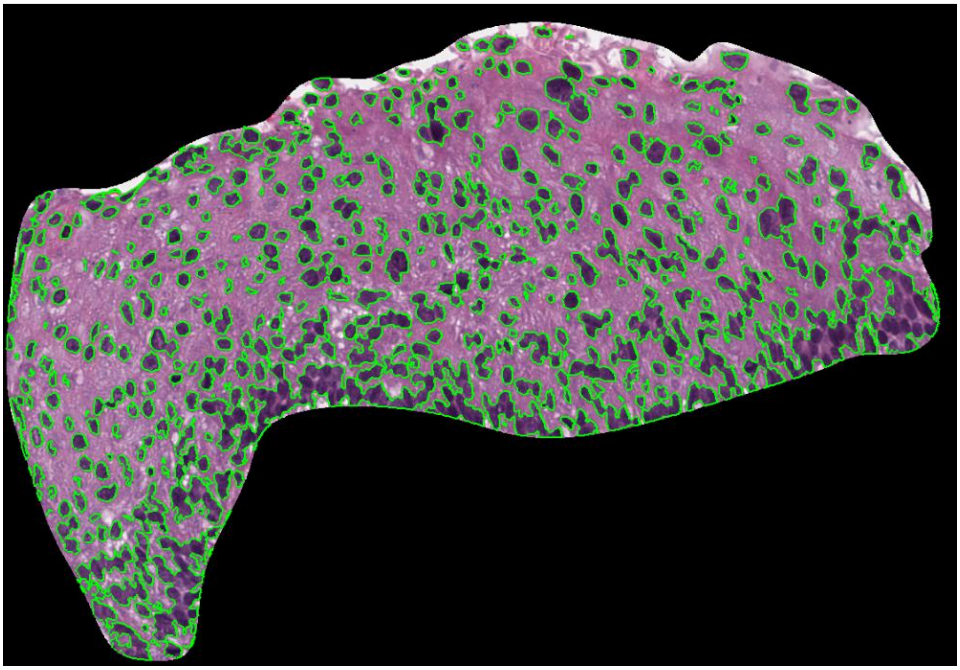

Figure 2.10 Detected nuclei boundaries marked in green

The color quantization technique using the K-means++ algorithm is applied on all images of the 71-image set from the database. The overall accuracy of nuclei segmentation was 93.67%, as shown in Table 2.1.

Table 2.1 Nuclei segmentation results using color quantization technique

Total no. of Nuclei	TP	FP	FN
75732	74468	1264	3450

The accuracy is calculated by manually obtaining the true positive (TP) (i.e., the number of nuclei successfully detected), false positive (FP) (i.e., the number of nuclei not

detected), and true negative (FN) (i.e., the value incorrectly indicating the absence of nuclei). Finally, the accuracy (μ) is given by [8]

$$\mu = \frac{TP - (FP + FN)}{TP}. \quad (1)$$

To segment the nuclei with the highest accuracy, a deep learning-based approach is employed by analyzing localized features through generation of superpixels and training convolutional neural networks.

3. DEEP LEARNING-BASED NUCLEI SEGMENTATION USING CONVOLUTIONAL NEURAL NETWORKS

Convolutional neural networks are inspired biologically with inputs from standard-sized digital image. These image arrays are made to convolve with the feature vectors defined as parameters to the convolutional neural network (CNN). The feature vectors are comprised of the weights that are modified for each iteration as the network learns from data used for training.

The primary goal of this project is to segment the nuclei in the epithelium of cervical cancer histology images by considering localized features instead of features from the whole image. This localized information is used to classify whether the segment contains nuclei or background. The CNN's use image vectors as inputs and learn different feature vectors, which ultimately solves the classification problem.

In order to make use of localized information, small chunks of images are obtained from the original image using a superpixel extraction method. Superpixel algorithms are devised to group the redundant pixels into regions to form a rigid pattern. Put simply, they capture the image redundancy and group the pixels to form clusters called superpixels.

3.1. PRE-PROCESSING

Before extracting superpixels, the original image is preprocessed using a Gaussian smoothing filter, which is used to blur the input image. This helps in reducing the noise present in the background of the input. Its impulse response is the Gaussian function, which decays rapidly. It is necessary to select narrow windows to avoid the decaying of the function. This function divides the image into its respective windows and applies the cost function. The Gaussian function that is applied on the input image is as follows:

$$g(x,y) = \frac{1}{2\pi\sigma^2} e^{-\left(\frac{x^2+y^2}{2\sigma^2}\right)}. \quad (2)$$

This is a two-dimensional Gaussian function. For implementation of this function, a built-in MATLAB function is used, which is also a two-dimensional Gaussian filter that uses a two-dimensional Gaussian smoothing kernel. The standard deviation can be user-defined, and if not defined, the default value is taken as two. The paper, the standard

deviation is taken as two. This filter is applied instead of a trimmed mean filter because the Gaussian filter takes 0.25 seconds whereas the trimmed mean filter takes 796.04 seconds. When the outputs of the algorithms were compared, the output using the Gaussian filter gave a better result than the output obtained using the trimmed mean filter. The luminance plane from the CIE color space of the smoothed image is obtained, and superpixels are extracted from the image.

3.2. SUPERPIXEL EXTRACTION

A simple linear iterative clustering (SLIC) algorithm is used to extract superpixels rather than other state-of-the-art methods because it is faster, more memory efficient, has better adheres to boundaries, and improves segmentation performance.

A labeled matrix of equal size to the original image is obtained as an output from the SLIC function. An epithelium mask is then applied on the labeled matrix to get rid of the unwanted region. The resultant matrix is again relabeled. Each superpixel formed is made to have at least 6 to 10 pixels, as shown in Figure 3.1.

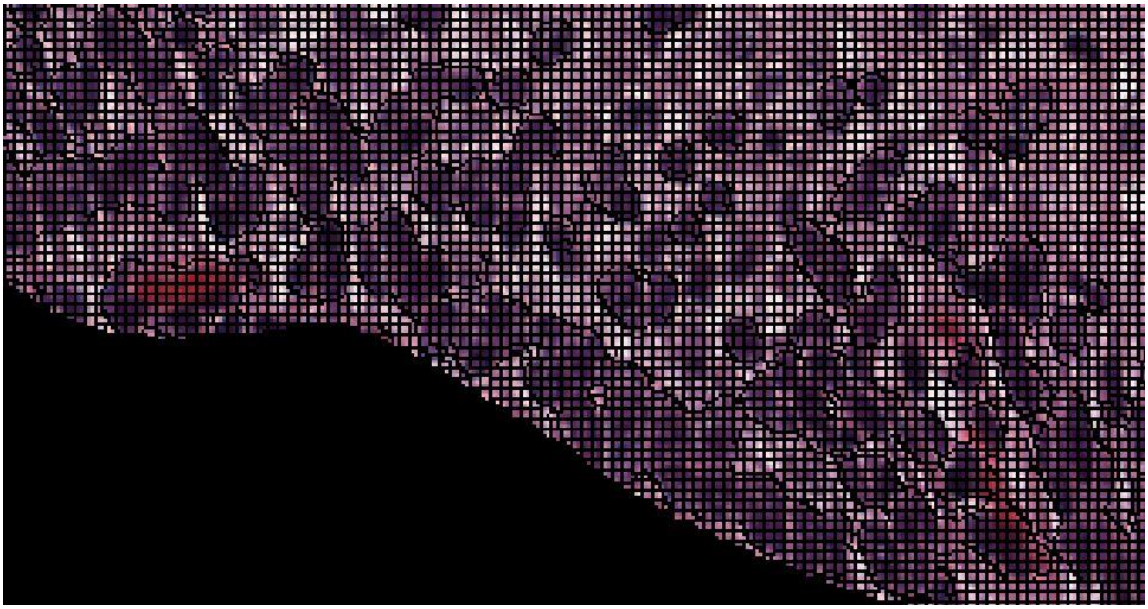

Figure 3.1 Original image with superpixels

The centroid of each superpixel is computed, and with respect to that centroid, patches of 16 x 16 x 3 image is formed as shown in Figure 3.2. A patch is said to be a part of the nuclei region if it has at least 0.1% of its area as nuclei. The nuclei region is given highest priority compared to the cytoplasm and background. The problem with generating 16 x 16 x 3 images from the superpixels at the edge of the image is solved by mirroring the image.

Figure 3.2 Generation of 16x16x3 RGB image from superpixel

Finally, 16 x 16 x 3 RGB input images are obtained from the superpixels of the original image. As more inputs are needed for deep learning, the original image is also rotated by 180 degrees and 16 x 16 images are extracted.

3.3. DATA GENERATION

Data generation is done carefully to prepare both training and test image data sets. For our experiment, a total of 12 images, six images each from the 71-image dataset and 62-image dataset are considered for the purpose of training the network. The images for training are carefully chosen so that the network understands how to handle different kinds of images. On observing the images from both 71-set and 62-set of data, it is clear that there are three kinds of images: images with light nuclei and light cytoplasm, images with darker nuclei and moderate cytoplasm, and images with darker nuclei and thicker cytoplasm as shown in Figure 3.3. A pair of each type of image were selected from both the sets, in order to train the convolutional neural network.

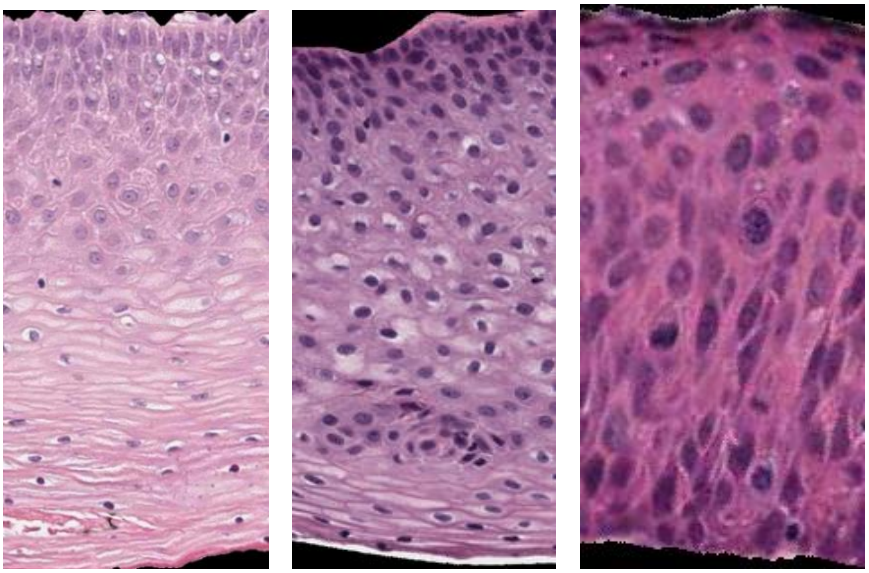

Figure 3.3 Images with lighter nuclei (left), darker nuclei with lighter cytoplasm (center), darker nuclei with thicker cytoplasm (right)

Classifying whether nuclei are present or not in the 16 x 16 x 3 patch is considered a binary classification problem. The target labels for each 16 x 16 x 3 patch is obtained from the binary nuclei masks that are already available in the database. Some of the portions of the nuclei masks are modified so that the target labels represent exact ground truth values. The extracted 16 x 16 x 3 patches are as shown in Figure 3.6. The label "0" denotes nuclei and the label "1" denotes background. A total of 377,012 patches are obtained using preprocessing steps as shown in Figure *3.4* Flowchart for generation of train data-setFigure *3.4* from 12 original images that comprise both nuclei and background.

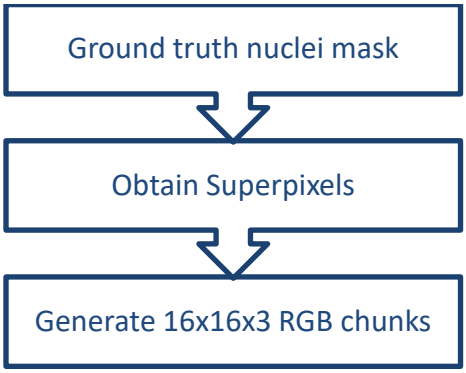

Figure 3.4 Flowchart for generation of train data-set

The test data is generated by preprocessing the image as mentioned in Figure *3.5*. The luminance plane is used to generate superpixels, and then 16 x 16 x 3 images are formed for each individual original image.

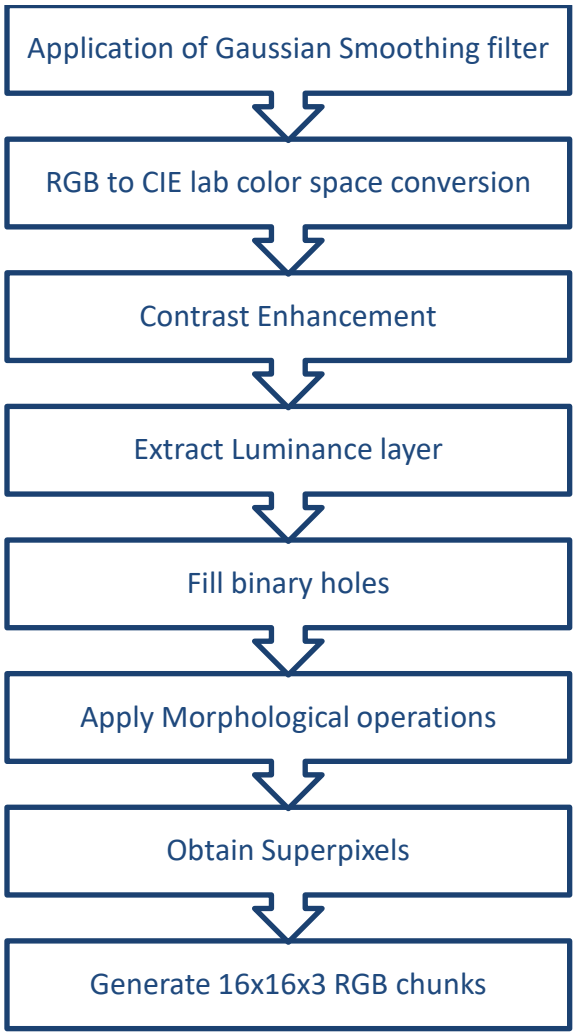

Figure 3.5 Flowchart for generation of test data-set

3.4. CONVOLUTIONAL NEURAL NETWORK

As a pre-step to train CNN network, all the small patches of images are converted to the HSV color plane and then the V-plane (value plane) is extracted. Before selecting the V-plane, various color planes are observed manually and are also used to train the network. The V-plane and the L-plane (luminance plane) gave promising results. The V-plane is considered for this experiment, as shown in Figure 3.6.

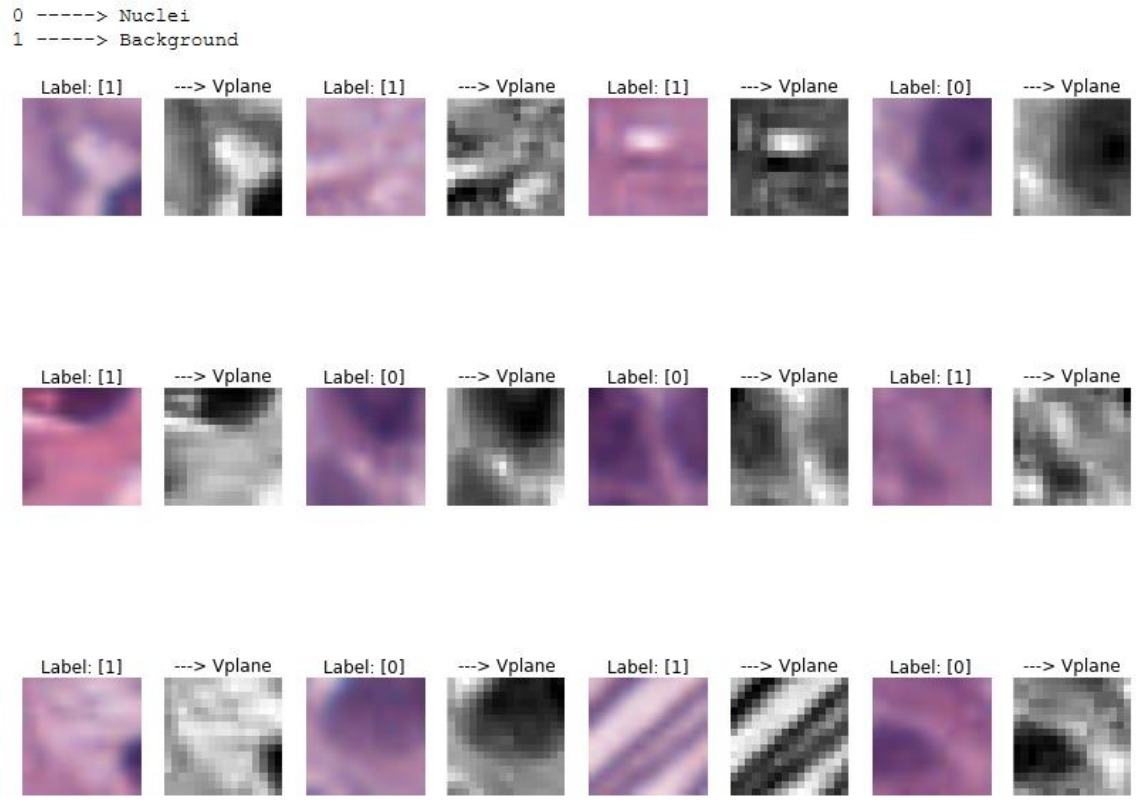

Figure 3.6 Samples of 16 x 16 x 3 RGB images and their 16 x 16 V-plane images

In order to classify the presence of nuclei, the convolutional neural network is trained with the features that were generated by convolutional layers using raw pixel input data. The first stage was a shallow CNN with one convolutional layer and a following max pool layer. Two images were considered for a quick quality check was completed. A remarkable improvement in the validation accuracy was observed when a deep CNN network architecture was considered with multiple convolutional, max pooling, and dropout layers. The architecture from Figure 3.7 has three consecutive convolutional, max pooling, and dropout layers at the beginning of the network and three regular neural networks (dense layers) at the end of the network. This produced a 96% validation accuracy on two input images. Later, 10 more images were included to make the network learn to classify nuclei in different environments, as shown in Figure 3.3.

Also the obtained data set of inputs and target labels are used to train CNN's with different architectures and the following architecture (Figure 3.7) gave best results with higher validation accuracy on test images that were part of the training data.

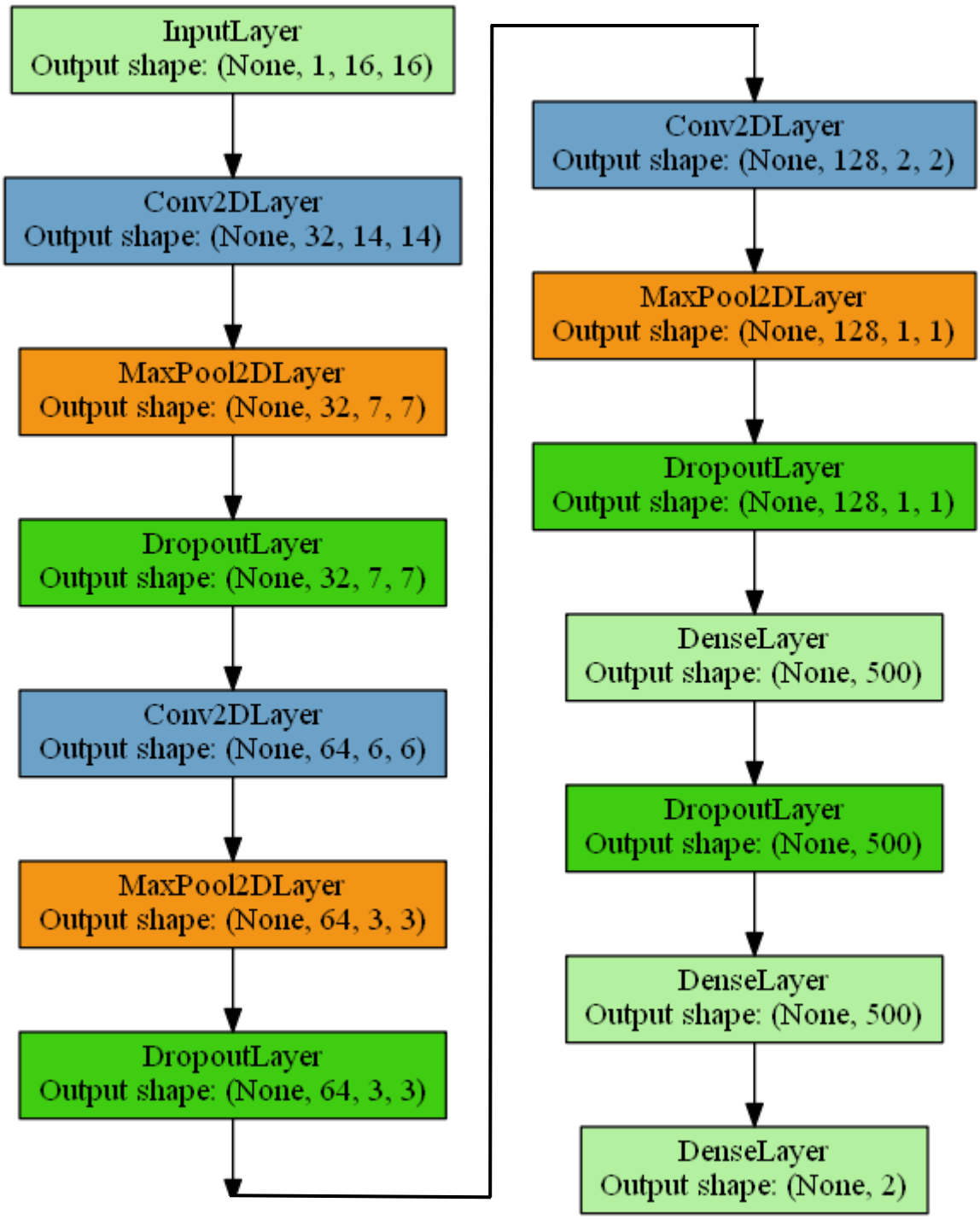

Figure 3.7 CNN architecture

The train dataset is used to fit the CNN model. A 20% of the train dataset is used as a validation dataset, which is helpful to estimate the prediction error for best model selection. Categorical accuracy is computed between prediction and targets produced from the validation dataset. An adaptable learning rate $\in$ (0.0001, 0.03) and momentum in the range $\in$ (0.9, 0.999) are applied to the network while training for 2000 epochs. The architecture produced a validation accuracy of 90.52% at the end of the 2000th epoch as shown in the Table 3.1.

Table 3.1 Training history

Epoch	Train_Loss	Valid_Loss	Valid_Accuracy	Duration
1	0.405191	0.365318	0.847796	80.49975
2	0.339616	0.336454	0.85873	80.4548
3	0.327013	0.315634	0.866628	80.45272
4	0.319594	0.307507	0.872228	80.4542
5	0.314946	0.304915	0.872022	80.44572
6	0.31122	0.320003	0.863062	80.44613
7	0.306248	0.31397	0.865538	80.45538
…	…	…	…	…
1995	0.233639	0.233551	0.90525	80.40883
1996	0.232405	0.23718	0.902701	80.42856
1997	0.23113	0.233445	0.902878	80.42384
1998	0.231225	0.231078	0.905015	80.42907
1999	0.229122	0.230274	0.904292	80.43345
2000	0.229737	0.232072	0.905265	80.44478

From the Table 3.1, the error on the training set is denoted as training loss. Validation loss is the error as a result of running the validation set through the previously trained convolutional neural network. Figure 3.8 represents a drop in training and validation error as the number of epochs increase. This is a clear indication that the network is learning from the data that is given as an input to the network.

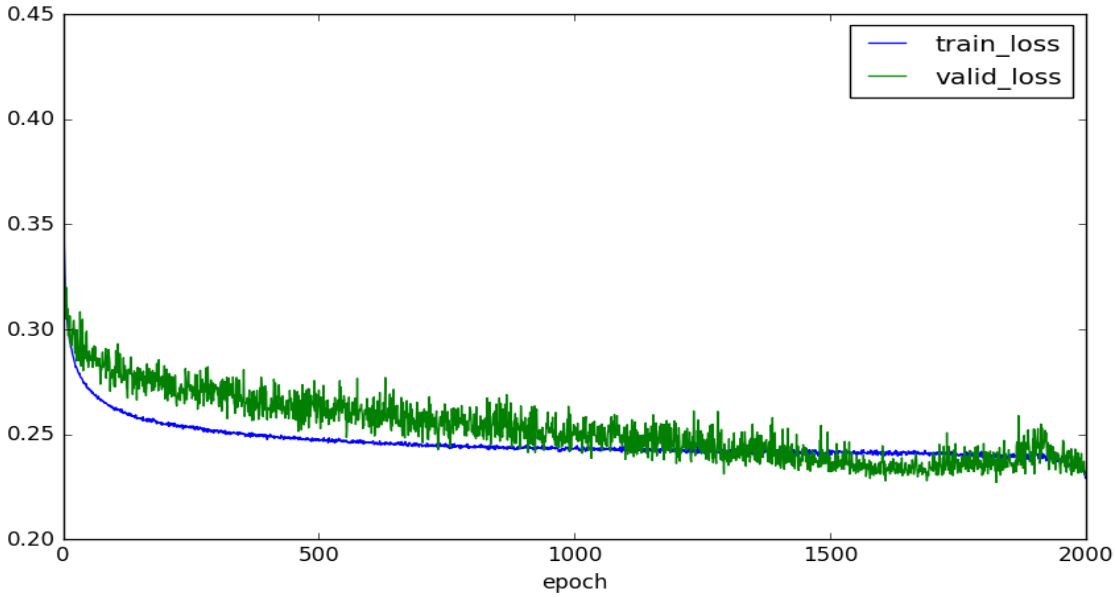

Figure 3.8 Train loss and validation loss vs number of epochs

The tabular data is represented graphically in Figure 3.8 and Figure 3.9. The drop in error rate with a train/valid loss ratio of approximately 1 is a desirable result denoting successful training of the network without overfitting.

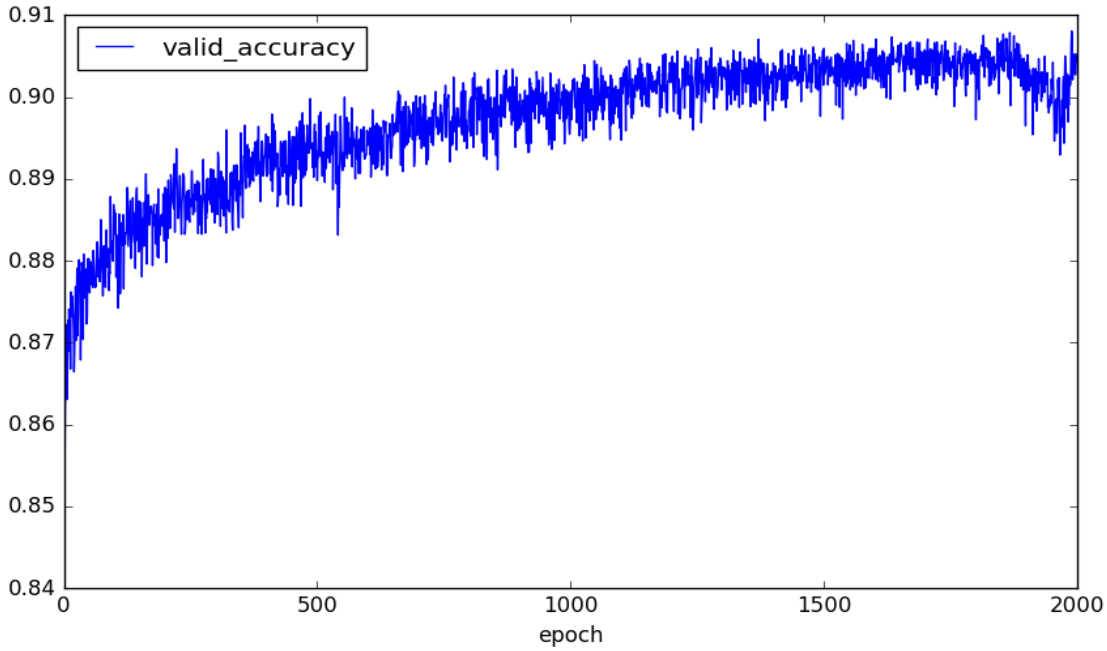

Figure 3.9 Validation accuracy vs number of epochs

Figure 3.10 shows all the 32 x 3 x 3 convolutional feature vectors of the first layer obtained as a result of the trained network. Figure 3.11 represents the result of the convolution of the feature vectors with the 16 x 16 image producing a 32 x 14 x 14 image.

Figure 3.10 32 x 3 x 3 Convolutional filters in the first layer

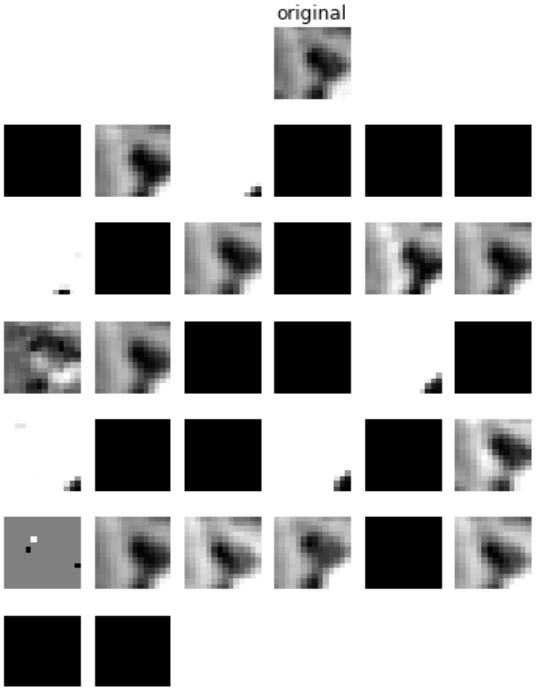

Figure 3.11 32 x 14 x 14 convolved output

The trained network model is saved along with the weights and filter coefficients. This saved model is loaded back to test on the remaining images of the 71-image dataset and the 62-image dataset to classify the image based on the presence of the nuclei. The location of every superpixel extracted from the original image is saved as a labelled image. The results of classification are mapped with the labeled image to finally obtain a binary nuclei mask from the corresponding original image. Figure 3.12 gives a visual outlook of the nuclei mask generated, and Figure 3.13 shows the nuclei boundaries marked in green obtained using its corresponding nuclei mask.

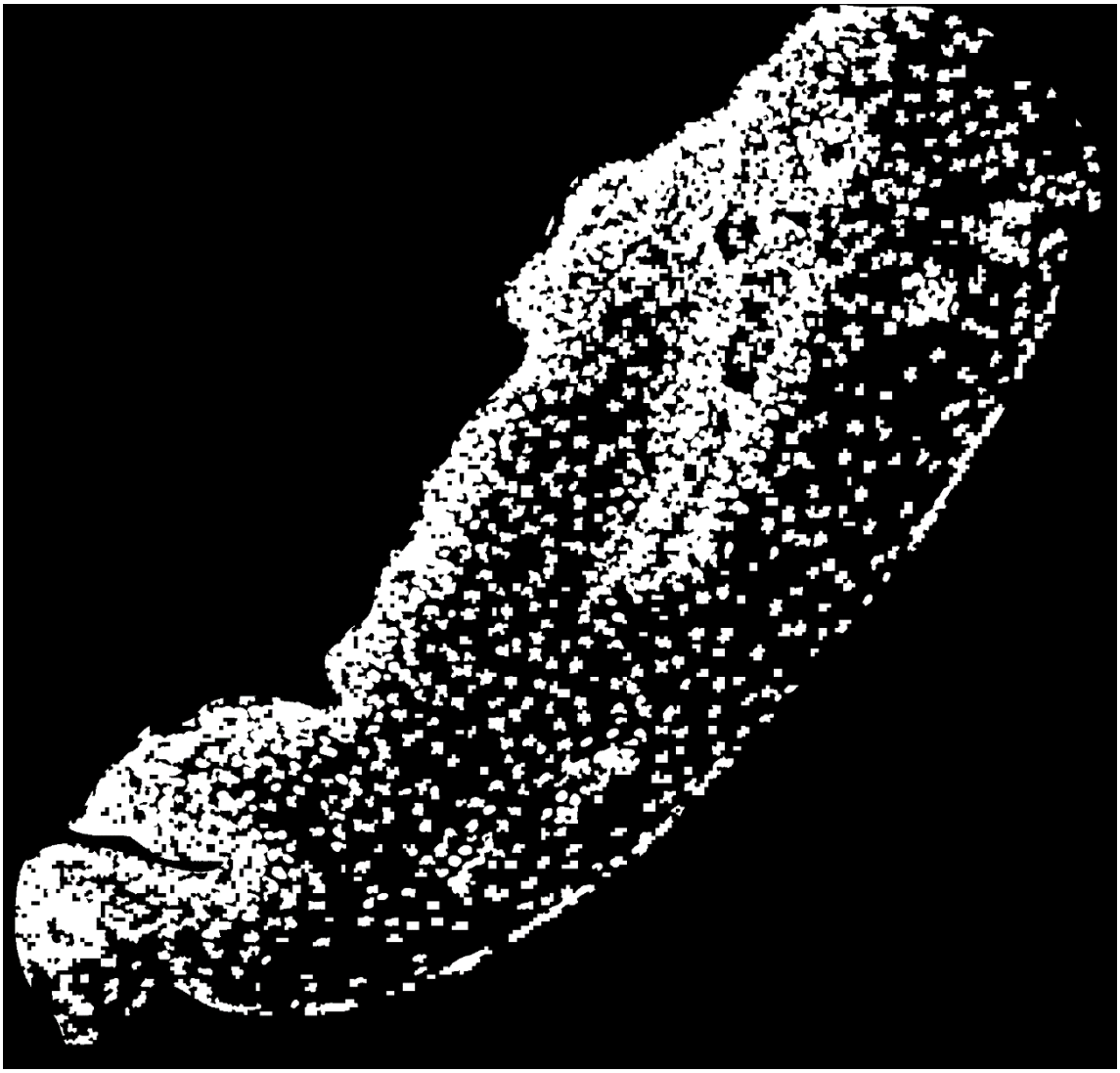

Figure 3.12 Generated nuclei mask

Figure 3.13 Boundary generated from Figure 3.12 superimposed on the original image

The deep learning algorithm is applied to both the 71-image dataset and the 62-image dataset with overall segmentation accuracy of 94.81% and 92.93%, respectively. Finally, the overall segmentation accuracy of the combined set is 93.87%.

The accuracy is calculated by manually recording the True Positive (TP) (i.e., the number of nuclei successfully detected), False Positive (FP) (i.e., the number of nuclei not detected), and False Negative (FN) (i.e., number of objects formed but are not nuclei). Finally, the accuracy (μ) is given by [8]

$$\mu = \frac{TP - (FP + FN)}{TP}. \tag{3}$$

Table 3.2 Nuclei segmentation results using the deep learning approach

Data-set	Total no. of Nuclei	TP	FP	FN	μ
71 set	75733	74353	1380	1565	96.04%
62 set	33594	31896	1698	510	93.07%
Combined set	109327	106249	3078	2075	95.15%

It is observed that if smaller size superpixels are considered, that is, the more the localization is done, the better the produced nuclei masks. Also, deeper CNN has shown remarkable classification results when compared to a shallow CNN.

4. REAL-WORLD PILL SEGMENTATION BASED ON SUPERPIXEL MERGE USING REGION ADJACENCY GRAPH

According to the National Library of Medicine [9], unidentified and misidentified pills present a challenge to patients, family members, and health professionals. Misidentified pills constitute a safety hazard. In the United States, nine out of 10 people over age 65 take more than one prescription pill, which may increase the chance of pill misidentification. This can lead to adverse drug events (ADE). This situation calls for automatic pill identification, enabling anyone to easily verify whether a pill with different size, shape, imprint, or color is a generic equivalent to the drug he or she was already taking. In an era of increasing polypharmacy and widespread use of 7-day pill dispensers, rapid and accurate automatic pill identification has lifesaving potential.

During the last decade, the improvement in computational power and digital camera technology has facilitated advances in machine vision research, yielding significant progress in automation of medical and industrial computer vision systems. Automatic identification of prescription drugs is now an increasingly important biomedical research topic.

Large prescription drug databases are now available to researchers. These databases include the National Library of Medicine (NLM) Pillbox database [9], DailyMedPlus [10], WebMD [11], and Drugs.com [12]. These resources provide various features of a pill, where users can manually access information on pill size, color, shape, and imprint to allow pill identification [13]. However, identification by manual website access is error prone and time-consuming. There is a need for an automatic pill identification system that is fast, reliable and easy to use.

Segmentation is the first and most critical step in the pill identification process. Segmentation isolates the pill from the background, enabling accurate analysis of the pill features. The images in Figure 4.1 are typical of pill images used in this project. These are examples of consumer-quality images provided by the NLM Pill Image Recognition Challenge 2016. Simple thresholding on the images in Figure 4.1 leads to significant segmentation errors, due to shadows and uneven lighting. These challenges are not present in the reference pill images shown in Figure 4.2 provided by the NLM challenge.

Figure 4.1 Consumer-quality pill images

The main objective of the NLM Pill Image Recognition Challenge was to use computer vision algorithms to rank lower-quality consumer images of prescription pills after training with higher-quality reference images as shown in Figure 4.2. These freely available high-quality digital images and associated data [14] were generated by NLM as part of the Computational Photography Project for Pill Identification. Although this challenge provided progress toward automatic pill identification, there is as yet (Fall 2016) no reliable and accurate automatic pill identification technology available.

The consumer-quality images, as shown in Figure 4.1, have issues such as low illumination, noisy background, and pill shadows, all of which pose great challenges in pill segmentation. When pill images include a noisy background, feature extraction algorithms can determine false features. Hence, there is a need to develop a segmentation algorithm to reduce these problems.

The proposed clustering segmentation algorithm includes three important steps [15]. Initially, pre-processing is done to over-segment the pill images by obtaining superpixels based on the modified k-means clustering algorithm. Secondly, a region

Figure 4.2 Reference pill images

adjacency graph is obtained from the over-segmented pill image to merge the regions within a certain threshold. Finally, various post-processing steps are applied to obtain the desired mask.

The goal of this paper is to accurately segment consumer-quality pill images captured using commonly available digital cameras and smartphones. After successful segmentation of the pill, in future work, features like shape, imprint and color will be extracted. These features help to compare, correlate and rank the consumer- quality images using the high-quality reference images.

4.1. METHODS

The main objective of the paper is to segment consumer-quality pill images that are affected by background noise and shadows. Once the pill is isolated, feature extraction is more reliable.

The proposed algorithm initially smoothens the image to reduce noise using a Gaussian smoothing filter. The simple linear iterative clustering (SLIC) algorithm [16] algorithm is then applied to generate superpixels. The resultant image is converted into a region adjacency graph and thresholded to merge the superpixels. A final binary mask is

obtained by thresholding color planes, applying an opening operation, filling holes, and applying a convex hull. A bounding box is applied to obtain only the segmented pill region.

4.1.1. SLIC Superpixels. The pre-segmentation of an image is a crucial step before applying region adjacency graphs. This step includes the generation of superpixels. Superpixels are a group of pixels that share similar characteristics with their neighboring pixels. They capture the image redundancy and subsequently reduce complexity in performing further image processing tasks. There are various approaches to generate superpixels [17][18][19][20][16]. This paper uses the SLIC algorithm to generate superpixels because it is faster, more memory-efficient, and has better boundary adherence than its predecessors. A detailed step-by-step procedure of the SLIC algorithm is provided in Achanta et al. [16].

The pill image is initially pre-processed using a Gaussian smoothing filter with standard deviation 2. The SLIC algorithm, which generates superpixels based on k-means clustering [21], is applied. The search space in the SLIC algorithm is limited to a specific region around a cluster centroid. This reduces the number of distance calculations, which in turn reduces the complexity and run time. It also considers a weighted distance approach by combining both color and spatial proximity. These features allow the algorithm to outperform existing state-of-the-art superpixel methods. The search is done for 10 iterations after initializing the cluster centroids. This generation of superpixels may be regarded as an over-segmentation process.

The output is a labelled image, as the algorithm assigns a unique label for each superpixel. An average color value of all pixels in a superpixel is calculated and assigned to the respective superpixel, as shown in Figure 4.3.

Formally, let μ_R denote the mean of a set of RGB color pixels $p_0, p_1, p_2, \dots, p_N$ in region R, as given by Equation (1):

$$\mu_R = \frac{1}{N} \sum_{i=0}^{N} p_i \tag{4}$$

where N is the total number of pixels in that region.

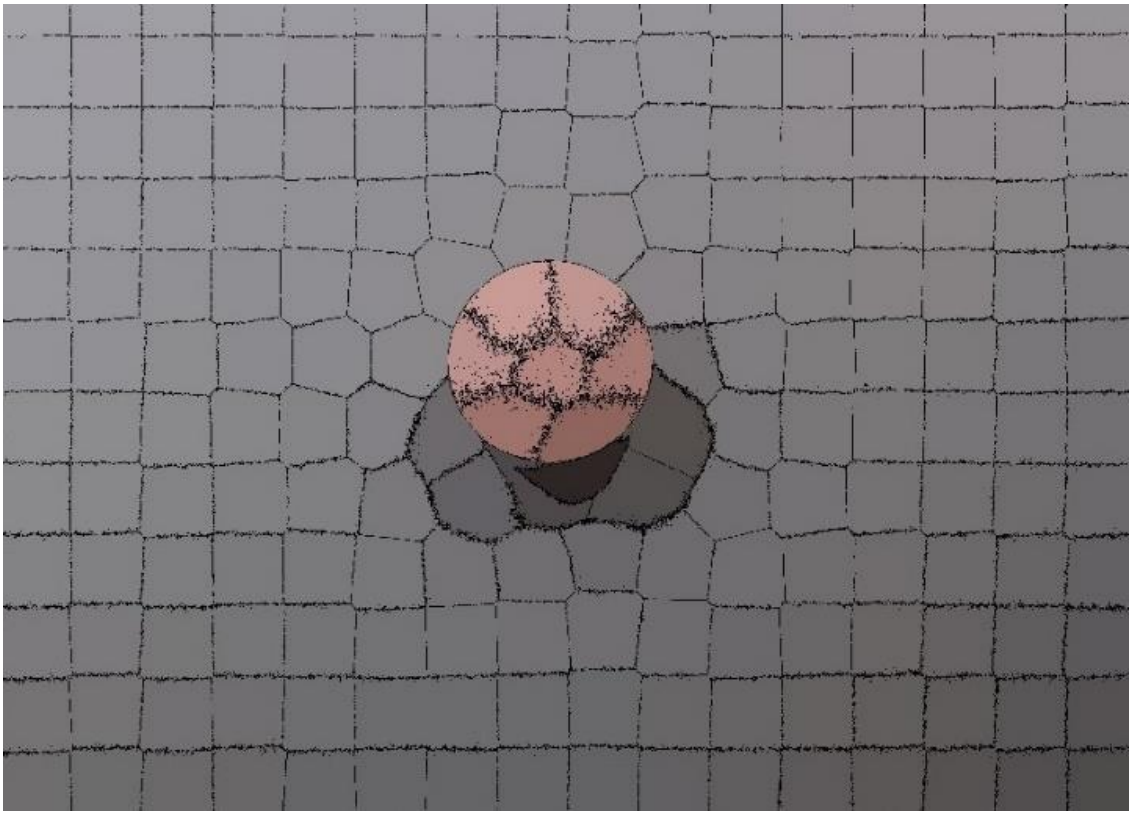

Figure 4.3 Pill segmented with superpixels with compactness factor = 12

4.1.2. Region Adjacency Graph. A region adjacency graph [22] is created as a step towards the merging of superpixels. The initial pre-segmentation (that is, the initial generation of superpixels) is crucial to create an associated adjacency graph. There is no loss of visual information in the pre-segmentation process. Pixels are only merged if they belong to the same superpixel region.

The over-segmented image is now considered as a graph. The centroid of each superpixel in the image is a node in the graph. All nodes in the adjacent regions are joined to form an edge as shown in Figure 4.4. This collection of edges is called the region adjacency graph.

The weight for the edge between two adjacent nodes [23] can be defined in various ways. The superpixels can be merged using these edge weights. As each superpixel is of uniform average color, the edge weights are defined by the difference of average color between the adjacent superpixel regions. The regions connected with a

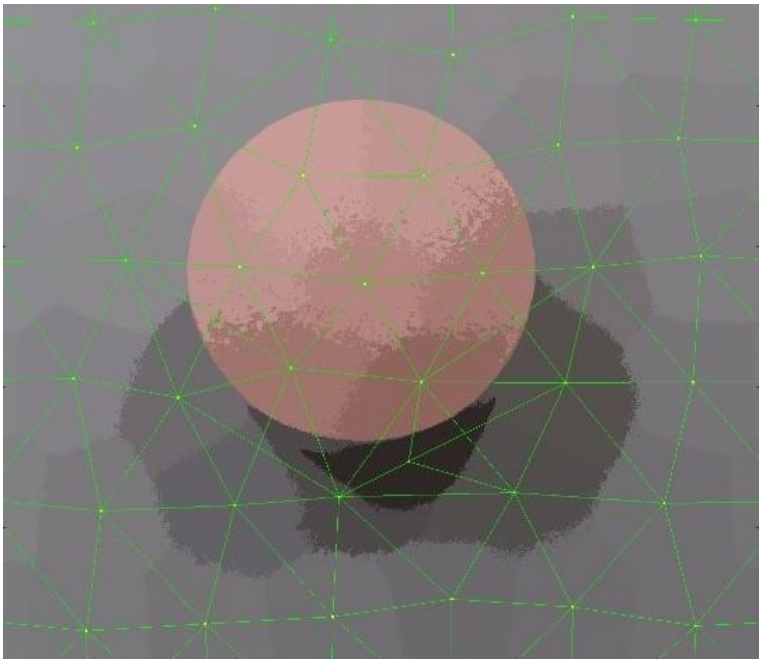

Figure 4.4 Labelled image (zoomed) with region adjacency

lower edge weight have similar color features and were merged using a threshold value of 29, determined empirically from a dataset of 30 random images from the provided consumer-quality images. The adjacent superpixel regions are merged if the edge weight is lower than the predetermined threshold value; if the edge weight is higher than the threshold value, the graph is cut as shown in Figure 4.5.

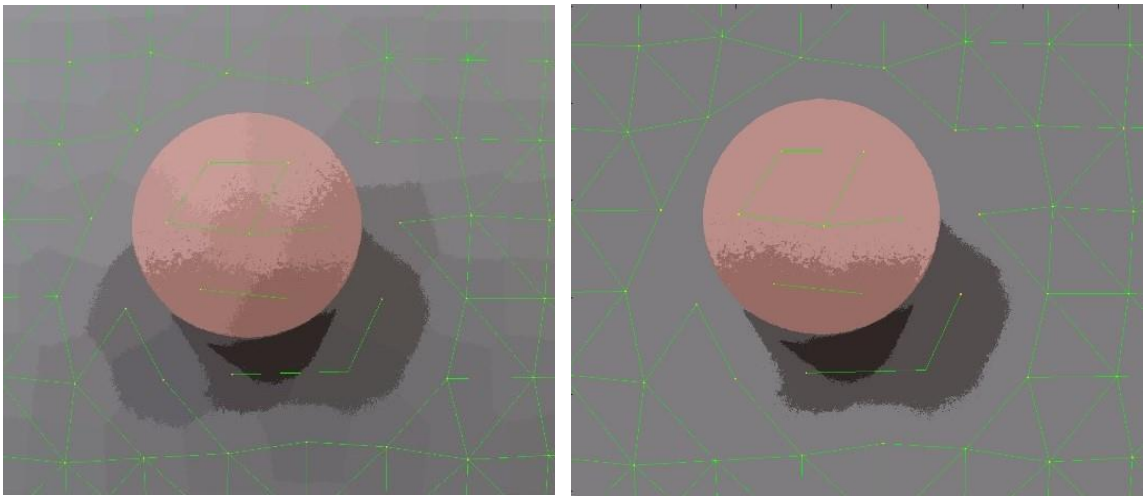

Figure 4.5 Superpixels with graph cut (left) and merged regions with graph cut (right) (zoomed)

As a result, a fully connected region adjacency graph (RAG) is divided into disconnected regions with threshold-cuts, as shown in Figure 4.6. The pixels of newly generated regions are assigned to the average color value of the merged regions. This substantially reduces segmentation complexity and results in easier generation of the pill mask.

4.1.3. Post-processing. The image resulting from merging superpixel regions by RAG thresholding is still affected by the shadows of the pill. The outer shadow needs to be merged with the background and the inner shadow should be merged with the object (pill).

Figure 4.6 Superpixels merge using RAG

When background color intensity is close to the pill color intensity, segmentation errors occur upon merging. To overcome this problem, a histogram of the image resulting after RAG thresholding is plotted, as shown in Figure 4.7. Since the background occupies most of the area in the image, the majority of the pixels share the same intensity level as

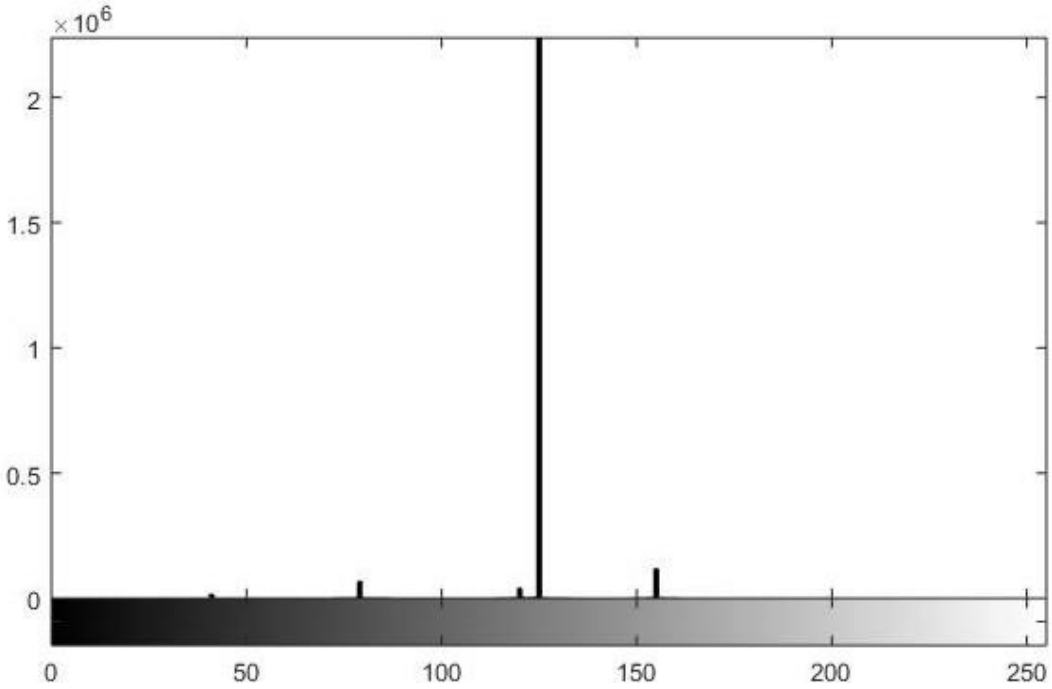

Figure 4.7 Histogram of image from Figure 4.6

that observed in Figure 4.7. The bin of the histogram with background pixels has the highest probability. All pixels sharing this most probable bin value are assigned to zero intensity. This overcomes the problem stated above.

On analyzing the color intensity values of various pill images, the red and blue planes contribute the majority of intensity changes from pill to its shadow. After reviewing 30 random consumer-quality images (previously used to determine the threshold for region connecting), threshold cutoff values of 105 and 83 were chosen for red and blue planes, respectively. An OR operation is applied to masks from both planes to generate a single binary mask.

A morphological opening (erosion followed by dilation) is then created to remove blobs of radius less than 9 pixels with a circular structuring element. Any holes in the mask are filled with a flood-fill operation. A final mask is generated by applying the convex hull operation on the filled mask as shown in Figure 4.8.

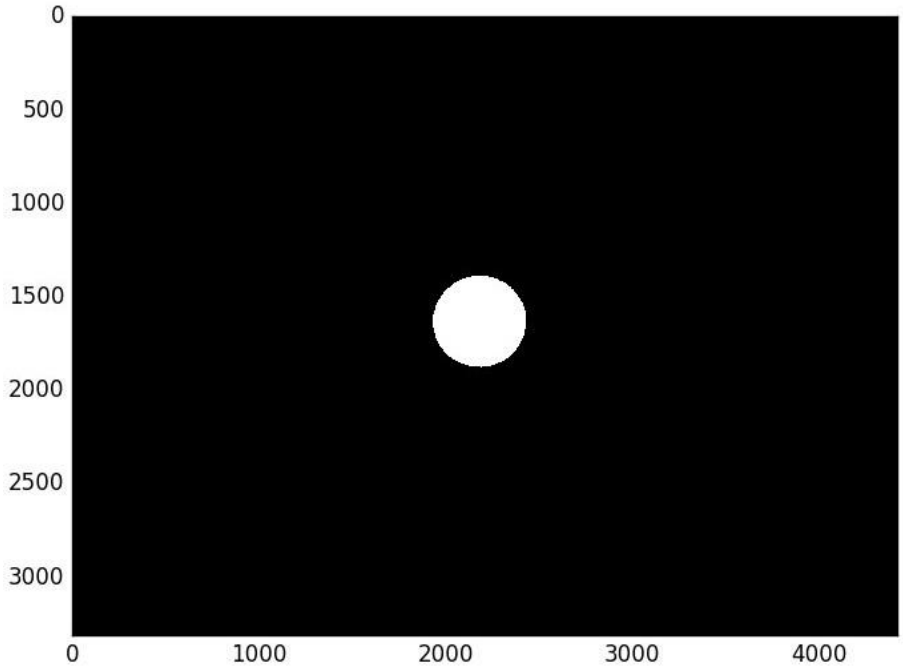

Figure 4.8 Binary mask of the pill

A distinct boundary along the edges of the pill is shown in the overlay image for this mask, Figure 4.9. A bounding box is applied to this mask to obtain the pill region as shown in Figure 4.10.

Figure 4.9 Boundary marked on the pill (zoomed)

Figure 4.10 Result of bounding-box

4.2. EXPERIMENTAL RESULTS

The proposed pill segmentation algorithm showed favorable accuracy results for the 5000 consumer-quality pill images provided by the NLM system. Since the algorithm uses a color segmentation approach, some of the pills with color similar to background color were completely merged with the background, resulting in a complete black mask. This is the primary limitation of the proposed algorithm.

The algorithm produced accurate segmentation results on the 2000 high-quality reference pill images as shown in Figure 4.12. These images are chosen as the benchmark for comparing the segmentation results of consumer-quality pill images.

The 5000 consumer-quality masked pill images were scored manually to analyze the accuracy of the segmentation with respect to segmentation of reference pill images. Results show accurate segmentation for 2243 pills, as shown in Figure 4.11 (left). For 1862 pills, some shadow is included along with the pill in the mask (Figure 4.11, center). The remaining pill images (17.9%) have false segmentation (Figure 4.11, right) due to the challenges mentioned above. In summary, the proposed algorithm produces acceptable segmentation accuracy for 82.1% of 5000 consumer-quality pills.

Figure 4.11 Bounding-box of segmented consumer pill images

Figure 4.12 Bounding-box of segmented reference pill images

The average time taken to run the algorithm (written in Python v2.7) on each pill image (of varying size with the largest being 2400 x 1600) on an Intel Core i5 2400 processor with 8 GB DDR3 RAM and a 512 MB AMD RADEON HD 6350 graphics card is 683.95 seconds. In order to make the segmentation proceed faster, a scaling factor is introduced and applied to reduce and resize the input image. The number of superpixels and the disk size for morphological operation are also reduced as the input image is scaled-down. But there is a trade-off with the quality of the mask generated as lower scaling factors are considered, as shown in Figure 4.13. This is shown in Table 4.1. The quality of generated binary masks on average is provided in Table 4.1, corresponding to 2243 of 5000 consumer-quality pill images with accurate segmentation accuracy.

Figure 4.13 Segmentation results with scale factor 1 (left), 0.4 (center), 0.1 (right)

The quality of the binary mask produced from each of those images for a varying scale factor ($i = 1.0, 0.9, 0.8, \ldots, 0.1$) is computed by Equation 5.

$$Q_i = \left(1 - \frac{|p_i - p_{1.0}|}{p_{1.0}}\right) * 100 \quad (5)$$

where Q_i is the segmentation quality of the binary mask, p_i is number of pixels in the object region of binary mask, and $p_{1.0}$ is the number of pixels in the object region of binary mask for a scale factor of 1.0. The speed factor was calculated as the ratio of the average run-time to process each image at a particular scale factor to that of the run-time to process the pill image with a scale factor 1.0. To provide the best segmentation results at a faster rate, a scale factor of 0.4 is considered to be the optimum value upon reviewing all the image masks from the dataset.

Table 4.1 Effect of scale factor on quality of binary mask and speed factor for individual pills

Scaling Factor	Speed Factor	Average Q value
1.0	1.00x	100%
0.9	1.11x	97.82%
0.8	1.66x	97.76%
0.7	1.95x	97.43%
0.6	2.93x	97.10%
0.5	4.12x	96.67%
0.4	**6.19x**	**98.46%**
0.3	10.30x	94.07%
0.2	19.08x	89.80%
0.1	40.30x	83.19%

5. CONCLUSIONS

The proposed method of nuclei segmentation through color quantization using K-means++ clustering has shown remarkable segmentation results, but with a few limitations. The histology images with lighter nuclei presented some difficulties in detection and segmentation. Images with a thicker cell wall posed a challenge, as they were detected as a part of nuclei mask.

The deep learning-based nuclei segmentation approach has shown improved segmentation results as compared to the formerly proposed method. Using deep learning, the full-size image is over-segmented by generating superpixels, which made the convolutional neural network (CNN) learn the localized features better in the training phase. The trained model is finally applied on the whole dataset, and the results prove that the deep-learning approach has outperformed the former color quantization approach.

The proposed method of merging superpixel regions using a region adjacency graph threshold-cut approach successfully segments consumer-quality pills with few limitations. Application of a resizing factor gave some promising results for algorithm speed, but with a trade-off in quality of mask.

Although the process has eliminated the background noise and produced excellent results for most of the pills and capsules; the shadows caused by pill illumination is still a challenge for some pills. Pills with similar background color also pose a great challenge in boundary determination. Finding an adaptable solution that works for all 5000 pills is challenging. Further analysis needs to be done to achieve accurate segmentation for all the consumer-quality pills.

This project was originally developed as an entry to the Pill Image Recognition Challenge conducted by the National Library of Medicine. The 5000 consumer-quality image datasets were accessed from the NLM database. Future work corresponds to the extraction of various features that are crucial to match the given consumer-quality pill.

BIBLIOGRAPHY

[1] M. J. Mcauliffe, F. M. Lalonde, D. Mcgarry, W. Gandler, K. Csaky, and B. L. Trus, “Medical Image Processing, Analysis & Visualization In Clinical Research,” pp. 381–386.

[2] W. H. Organization, *Department of Reproductive Health and Research and Department of Chronic Diseases and Health Promotion Comprehensive Cervical Cancer Control: a Guide to Essential Practice*. Geneva, Switzerland: WHO Press, 2006.

[3] J. Jeronimo, M. Schiffman, L. R. Long, L. Neve, and S. Antani, *A tool for collection of region based data from uterine cervix images for correlation of visual and clinical variables related to cervical neoplasia*. 17th IEEE Symp. Computer-Based Medical Systems: IEEE Computer Society, 2004.

[4] V. Kumar, A. Abba, N. Fausto, and J. Aster, *The female genital tract in Robbins and Cotran Pathologic Basis of Disease, 9th ed.*, vol. 22. 2014.

[5] L. He, L. R. Long, S. Antani, and G. R. Thoma, “Computer assisted diagnosis in histopathology in Sequence and Genome Analysis: Methods and Applications,” *iConcept Press. Hong Kong*, vol. 15, pp. 271–287, 2011.

[6] Y. Wang, D. Crookes, O. S. Eldin, S. Wang, P. Hamilton, and J. Diamond, “Assisted diagnosis of cervical intraepithelial neoplasia (CIN),” *IEEE J. Sel. Top. Signal Process.*, vol. 3, no. 1, pp. 112–121, 2009.

[7] H. Palus and M. Frackiewicz, “New approach for initialization of k-means technique applied to color quantization,” no. June, pp. 205–209, 2010.

[8] M. Szénási, S., Vámossy, Z. & Kozlovszky, “Evaluation and comparison of cell nuclei detection algorithms,” in *16th IEEE International Conference onIntelligent Engineering Systems (INES2012)*, 2012, pp. 469–475.

[9] NLM, “National Library of Medicine : Pill Image Recognition Challenge.” 2016.

[10] Dailymedplus, “Medicos consultants.” 2016.

[11] WebMD, “Pill Identification.” 2016.

[12] Drugs.com, “Pill Identifier.” 2016.

[13] J. J. Caban, A. Rosebrock, and T. S. Yoo, “Automatic identification of prescription drugs using shape distribution models,” in *2012 19th IEEE International Conference on Image Processing*, 2012, pp. 1005–1008.

[14] C3PI, “Computational Photography Project for Pill Identification.” 2016.

[15] R. Xu and D. Wunsch, “Survey of clustering algorithms,” *IEEE Trans. Neural Networks*, vol. 16, no. 3, pp. 645–678, 2005.

[16] R. Achanta, A. Shaji, and K. Smith, “SLIC Superpixels Compared to State-of-the-Art Superpixel Methods,” *Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2281, 2012.

[17] Y. Y. Boykov and M. P. Jolly, “Interactive graph cuts for optimal boundary amp; region segmentation of objects in N-D images,” in *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, 2001, vol. 1, pp. 105–112 vol.1.

[18] J. Shi and J. Malik, “Normalized cuts and image segmentation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.

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

[20] P. F. Felzenszwalb and D. P. Huttenlocher, “Efficient Graph-Based Image Segmentation,” *Int. J. Comput. Vis.*, vol. 59, no. 2, pp. 167–181, 2004.

[21] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, “An efficient k-means clustering algorithm: analysis and implementation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 881–892, Jul. 2002.

[22] A. Tremeau and P. Colantoni, “Regions adjacency graph applied to color image segmentation,” *IEEE Trans. Image Process.*, vol. 9, no. 4, pp. 735–744, 2000.

[23] S. van der Walt, J. L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E. Gouillart, T. Yu, and the scikit-image contributors, “scikit-image: image processing in {P}ython,” *PeerJ*, vol. 2, p. e453, 2014.

VITA

Sudhir Sornapudi was born in Visakhapatnam, Andhra Pradesh, India. He attained his Bachelor of Technology degree in Electronics and Communications Engineering from JNTUK – University College of Engineering Vizianagram, India (May 2014). He then pursued his Master of Science (thesis) degree in Computer Engineering as a graduate student at Missouri University of Science and Technology and graduated in May 2017. His interests mainly include computer vision, machine learning and deep learning.