

Friedrich-Alexander-Universität Erlangen-Nürnberg
Artificial Intelligence in Biomedical Engineering
W3-Professur für Image Data Exploration and Analysis
Prof. B. Kainz
W3-Professur für Computational Imaging
Prof. F. Knoll

Medizintechnik II

Projektarbeit
Sommersemester 2024

Fundamental Image Processing Techniques in Biomedical Imaging

November 24, 2024

Studierende*r:

Muhammad Muaviya Ijaz

23454875

Contents

1	Introduction	3
2	Method	4
2.1	Thresholding and Illumination-Correction	4
2.2	Image Segmentation	6
2.3	Otsu's Method	8
2.4	Edge-detection	12
2.5	Canny edge detection	15
3	Conclusion	18
	References	20

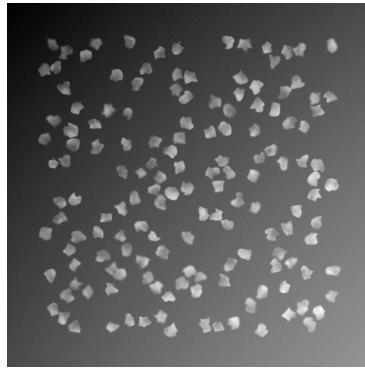


Figure 1 Grayscale Image showing clusters of cells

1 Introduction

In the progressing field of biomedical engineering, **Image Processing** plays a vital role in enhancing medical diagnostics and research. Basic methods such as thresholding, segmentation, and edge detection are crucial for improving and examining biomedical images, like those showing cellular structures. These fundamental techniques are essential instruments for biomedical engineers, allowing them to enhance image quality and obtain valuable data, consequently influencing the precision of diagnoses and research results.

This report goal is to present **key image processing techniques** that can be applied directly to cellular imaging, offering practical knowledge. This study underscores the importance of these methods in analyzing biomedical images by linking theoretical concepts to practical applications. The primary emphasis is on basic methods such as thresholding, Otsu's algorithm, segmentation and its evaluation, and edge detection, serving as the foundation for advanced image analysis tools like Fourier Transform filtering, Transfer learning, and Convolutional Neural Networks (CNNs). Having a strong grasp of these fundamental skills is essential for advancing to more complex techniques and creating innovative algorithms in the field of biomedical imaging.

The efficient processing and analysis of biomedical images to extract meaningful and correct information is the **primary goal** of this report. In particular, the report concentrates on describing and evaluating various image segmentation algorithms for the purpose of analyzing and identifying cellular features as portrayed by the provided cells image, shown in Figure 1. The **main research question** is: "How can basic image processing techniques be applied to biomedical images, such as those depicting cellular structures, to enhance the efficiency and accuracy of segmentation?" This, in turn, will enhance the overall analysis and interpretation of biomedical data. **ChatGPT** is used as a writing assistant in this report to enhance paragraph-level grammar, spellchecking of the content, and improve the clarity of explanations. However, the content, analysis, and conclusions presented are entirely by our own research [OpenAI, 2024].

Figure 1 shows an image of cluster of cells. In the following parts, we will explore the precise techniques as mentioned before in the report. The evaluation of these methods' efficacy will be based on how they are used with this original image of cells.

2 Method

2.1 Thresholding and Illumination-Correction

In this section, we explain the basic techniques like **Basic Thresholding** and **Illumination Correction** in microscopy images. These techniques are crucial for enhancing image analysis by transforming grayscale images into binary images and fixing the problem of uneven illumination, a frequent occurrence in microscopy. The procedure includes transforming an 8-bit grayscale picture into a binary image using a threshold value set by the user. Moreover, a supplementary measure is taken to address the problem of uneven illumination, a frequent issue encountered in microscopy imaging. To demonstrate these techniques, we take the example of the image of cells again, as shown in Figure 1.

Thresholding is a basic method in image processing which changes a grayscale image to a binary image. Thresholding aims to divide the image into black and white regions by assessing pixel intensity values, distinguishing between foreground and background. This technique is particularly beneficial in binarization of the image, facilitating further image analysis like segmentation or object detection [Gonzalez, 2009].

Basic thresholding involves the user selecting a specific intensity threshold, t , during the process. The intensity value, $I(x, y)$, of each pixel in the grayscale image is checked against the threshold t . After comparing, the pixel is categorized as foreground or background. The outcome is an image that is binary. The **mathematical representation of using thresholding** for image binarization, as mentioned and discussed in the survey [Sahoo et al., 1988], can be expressed as:

$$T(x, y) = \begin{cases} b_1 & \text{if } I(x, y) > t \\ b_0 & \text{otherwise} \end{cases} \quad (1)$$

where x and y are the coordinates in the Image I for a specific pixel value and variables b_1 , b_0 represents values for white and black color respectively. If the pixel value is greater than the threshold value t , that pixel location is set value of 255 i.e white, otherwise 0 i.e black.

Another issue which happens to be there in microscopy images is of **Uneven Illumination**. It can result in incorrect analysis because of gradients or vignetting effects. In order to fix this issue, we adjust the image's brightness by dividing it by a blurred version that signifies the lighting variation. Mathematically speaking, this is represented in the form of the following equation:

$$C(x, y) = \frac{I(x, y)}{B(x, y)} \quad (2)$$

where $C(x, y)$ is the corrected pixel value, $I(x, y)$ is the original pixel value, and $B(x, y)$ is the blurred image. The blur, achieved with a large-radius Gaussian filter, captures the illumination gradient, effectively correcting the image. The advanced form of this method is also cited in this article [Smith et al., 2015] where they employed a model-based approach that handled complex and varying illumination variations across approximate 11 datasets.

Ensuring accurate analysis in microscopy is essential through illumination correction to prevent gradients and vignetting that can obscure details. Vignetting decreases brightness or saturation around the edges of an image, usually caused by problems with the light transmission from the microscope to the camera. Distortions of this kind can result in inaccuracies in quantitative measurements and impact tasks such as image segmentation and mosaicking. It is crucial to rectify these artifacts to prevent inaccurate outcomes in medical diagnosis. Recent techniques, such as a Fully Convolutional Network (FCN) with feature encoding, decoding, and detail supplementation, have shown to be successful. This method, when applied to pathological cell images, surpasses conventional techniques, enhancing image mosaicking and visual quality [Wang et al., 2021].

The following figures showcase example of microscopy image before and after thresholding done on Figure 1, with and without illumination correction, based on calculations of above mentioned Equations 1 and 2, implemented in our project.

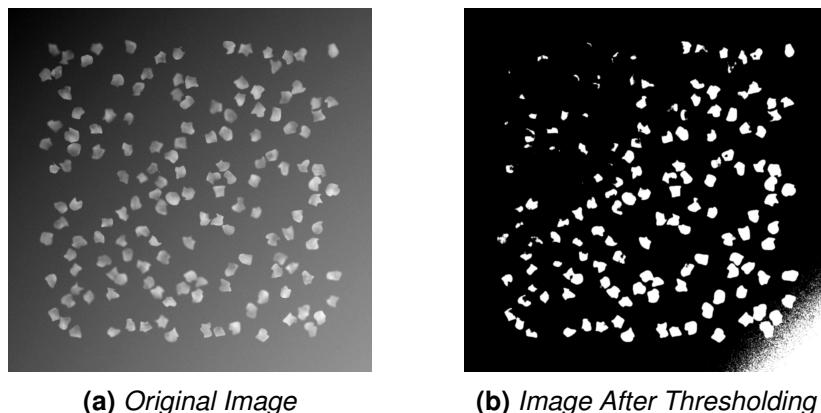


Figure 2 Comparison of images before and after thresholding with $t = 135$

Figure 2 compares the original microscopy image, shown in Figure 2a with its thresholded version, shown in Figure 2b using a threshold value of $t = 135$. The thresholding was applied without prior illumination correction, resulting in inaccuracies, particularly near the image edges due to uneven lighting in the original image.

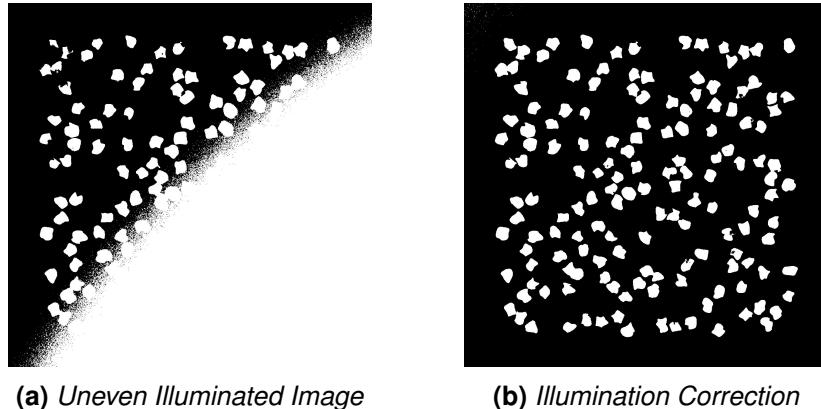


Figure 3 Comparison of images before and after illumination correction with $t = 80$

Figure 3 demonstrates the effect of the **correctIllumination** method. The original image suffered from uneven illumination as shown in Figure 3a, which caused inaccurate thresholding. After applying the illumination correction, the image intensity was normalized, so that the lighting is evenly distributed across the image, resulting in a more accurate thresholding output, shown in Figure 3b.

2.2 Image Segmentation

In this section, some **key evaluation metrics** for assessing the results of Image Segmentation in Section 2.1 will be discussed. We will discuss the metrics such as True Positives (TP), True Negatives (TN), False Negatives (FN), and False Positives. True Positives (TP) refer to the number of correctly identified pixels that belong to the object. The number of correctly identified pixels as background is shown by True Negatives (TN). If background pixels are incorrectly classified as part of the object, it leads to False Positives (FP), while if object pixels are mistakenly identified as part of the background, it results in False Negatives (FN).

Two key measures for evaluating a segmentation algorithm's effectiveness are Specificity and Sensitivity. These metrics are also used for interpreting diagnostic tests for physician's daily practice [Van Stralen et al., 2009]. Below are the given formulas:

$$\text{Specificity} = \frac{TN}{TN + FP} \quad (3)$$

$$\text{Sensitivity} = \frac{TP}{TP + FN} \quad (4)$$

When evaluating Image Segmentation, **Specificity** refers to the percentage of background pixels in the image that are accurately identified. The objective is to achieve a high level of specificity by reducing the number of false positives, making sure that background pixels are not classified as part of the foreground object, as mathematically calculated in Equation 3. Contrary to this, **Sensitivity** refers to the ratio of correctly

identified foreground pixels. A high-quality image segmentation algorithm decreases the number of false negatives by correctly identifying foreground pixels and not misclassifying them as background pixels, as mathematically derived in Equation 4. To summarize, these measurements offer a thorough grasp of how well the algorithm works, considering the importance of accurate object detection while also ensuring background is properly excluded. Elevated values for both of these measurements suggest a strong and dependable segmentation procedure.

To evaluate these metrics, we will use the image of cells as shown in Figure 1 , and then create six different segmentation by applying illumination correction and using various threshold values. Each segmentation will be compared against a reference image, with a focus on evaluating Sensitivity and Specificity. The referenced image which we will use for this task is shown below:

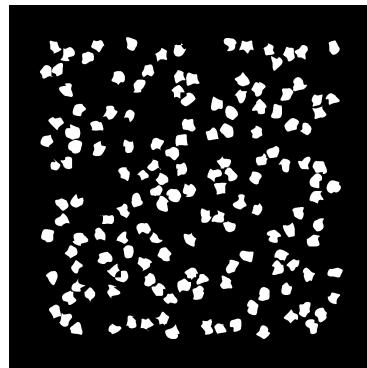


Figure 4 *Referenced Cells Image*

Figure 4 shows the image of referenced cells against which we will compare our segmentation results. After performing segmentation on Figure 1, we got the following results:

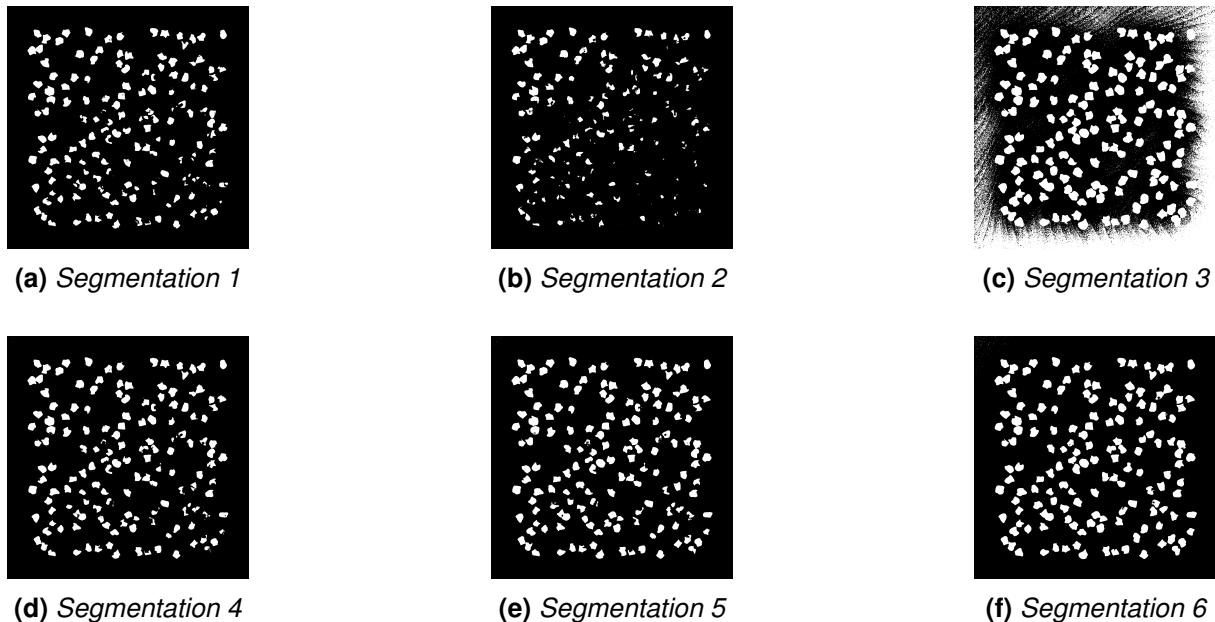


Figure 5 *Different segmentations of cells image*

Figure 5 shows the evaluation of different segmented images, produced from methods of illumination correction and basic thresholding techniques as mentioned in Section 2.1. Below are their results presented in the table:

Segmentation Figure	Threshold	Sensitivity	Specificity
5a	92	0.70	0.99
5b	105	0.40	0.99
5c	60	0.99	0.72
5d	88	0.78	0.99
5e	84	0.86	0.99
5f	74	0.97	0.99

Table 1 Sensitivity and Specificity Results

By comparing the segmented images in Figure 5 with the referenced cell image in Figure 4, we assessed Sensitivity and Specificity of these different segmentations. The outcomes of this evaluation are summarized in Table 1. Figure 5f provides the best balance between sensitivity (0.97) and specificity (0.99), making it the most effective overall for this dataset.

In addition to Sensitivity and Specificity, the **Dice Coefficient** and the **Jaccard Index** are also crucial measures for evaluating the quality of image segmentation. The Dice Coefficient, also known as the F1 Score, is useful in assessing overlap as it considers both precision and recall to measure the similarity between the segmented output and the ground truth. Alternatively, the Jaccard Index, also referred to as Intersection over Union, evaluates accuracy by calculating the proportion of the intersection to the union of expected and observed segments. These metrics have also been utilized to assess Skin Lesion [Setiawan, 2020].

2.3 Otsu's Method

Otsu's method is a popular thresholding technique in image processing, commonly used for image segmentation. Developed by Nobuyuki Otsu in 1979. This technique automatically determines the optimal threshold for separating an image's pixels into foreground and background. This is accomplished by reducing differences within the same class and increasing differences between the two pixel classes. Otsu's method calculates the threshold value by assessing the histogram of the image, which shows the spread of pixel intensities. Pixels below the threshold are considered background, while those exceeding it are labeled as foreground [Otsu, 1979].

The **main objective of Otsu's method** is to convert grayscale images into binary images by analyzing the histogram of pixel intensity to find the optimal threshold for separating the image into two groups. The method identifies every possible threshold and selects the one that maximizes the variance between classes, enabling the effective differentiation between foreground and background. Due to its worldwide scope and emphasis on differences rather than specific local features such as pixel intensity

lows or peaks, this technique is particularly effective in settings with changing lighting and noise levels.

Firstly, we have a normalized histogram h , where $h(i)$ denotes the likelihood of every pixel intensity i in the image. The technique utilizes the histogram to determine the ideal threshold by maximizing the inter-class variance. Listed below are the initial equations to start with:

$$P_1(\theta) = \sum_{i=0}^{\theta} h(i) \quad (5)$$

$$P_2(\theta) = 1 - P_1(\theta) = \sum_{i=\theta+1}^{L-1} h(i) \quad (6)$$

Equation 5 and 6 represent the probabilities $P_1(\theta)$ and $P_2(\theta)$ that shows the likelihood of a pixel intensity falling into the background or foreground class, respectively. $P_1(\theta)$ is the cumulative sum of the histogram values $h(i)$ from 0 to θ , representing the number of pixels below the threshold (background). $P_2(\theta)$ is the sum of the histogram values from $\theta + 1$ to $L - 1$, representing the number of pixels above the threshold (foreground), where L is the length of all pixel intensities shown in the histogram .

Then the mean intensities for both classes are calculated. The mean intensity values $\mu_1(\theta)$ and $\mu_2(\theta)$ are the mean intensities of the background and foreground classes, respectively. They are calculated as:

$$\mu_1(\theta) = \frac{1}{P_1(\theta)} \cdot \sum_{i=0}^{\theta} (i + 1)h(i) \quad (7)$$

$$\mu_2(\theta) = \frac{1}{P_2(\theta)} \cdot \sum_{i=\theta+1}^{L-1} (i + 1)h(i) \quad (8)$$

where $\mu_1(\theta)$ is the weighted sum of the intensity levels up to the threshold θ , divided by the probability $P_1(\theta)$, giving the mean intensity of the background. Similarly, $\mu_2(\theta)$ is the weighted sum of the intensity levels from $\theta + 1$ to $L - 1$, divided by the probability $P_2(\theta)$, giving the mean intensity of the foreground.

The inter-class variance $\sigma_B^2(\theta)$ is a measure of how distinct the two classes (foreground and background) are. It is calculated as:

$$\sigma_B^2(\theta) = P_1(\theta) \cdot P_2(\theta) \cdot (\mu_1(\theta) - \mu_2(\theta))^2 \quad (9)$$

Here, Equation 9 shows the formula for calculating $\sigma_B^2(\theta)$ representing the inter variance for a given threshold θ . The larger this variance, the better the separation between the two classes. $P_1(\theta)$ and $P_2(\theta)$ are the probabilities of the classes, and $\mu_1(\theta)$ and $\mu_2(\theta)$ are their respective mean intensities. The point where we get the maximum variance between the two classes defines our optimal threshold.

Although Otsu's method is frequently utilized, **it has some limitations**. Initially, it assumes that the image histogram displays bimodality, showing two distinct peaks that symbolize the foreground and background. However, the technique may not be effective if the histogram has a single peak or multiple peaks [Sezgin and Sankur, 2004]. The limitation is discussed in the referenced paper, as the authors aimed to evaluate how different image segmentation techniques perform on a variety of image categories. They found out that Otsu's technique is limited because it needs a bimodal histogram, which reduces its effectiveness on images with unimodal or multimodal distributions. Moreover, noise and non-uniform illumination can affect Otsu's method, changing the histogram and leading to poor threshold selection, possibly compromising segmentation accuracy [Huang and Wang, 1995]. The authors of this paper explore various methods for image thresholding in a specific example, assessing their performance under conditions of noise and uneven lighting. They specifically assess Otsu's method, noting that while it is effective in certain scenarios, its reliance on general histogram characteristics makes it vulnerable to noise and uneven lighting disruptions.

For the purpose of this report, Otsu's method was applied to segment the given image as shown in Figure 1 by determining the optimal threshold value, which was found to be **92**. This threshold effectively separates the image into two distinct classes: foreground and background. Specifically, pixels with intensities below 92 were classified as background, while those above this threshold were classified as foreground. The segmentation result demonstrates the method's ability to distinguish between different regions of the image based on pixel intensity distribution.

To illustrate this using Otsu's method, the following figure presents the resulted segmented image

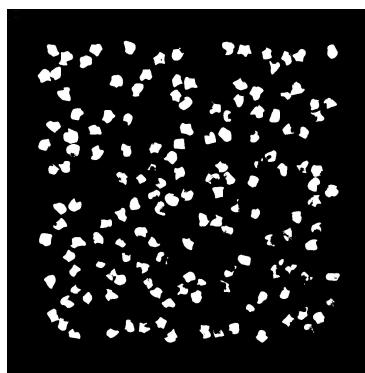


Figure 6 *Otsu Segmented Image with optimal threshold t = 92*

Figure 6 shows that after applying Otsu's method on the original image of cells, shown in Figure 1, we got the following result which was able to classify foreground and background pixels based on maximum intra class variance value, which was found at a **threshold value of 92**.

When addressing the problem of selecting an optimal threshold in otsu method, it is important to understand that why this threshold is selected on the integration rather than differentiation of the histogram as Otsu also mentions this in his original publication "An optimal threshold is selected automatically and stably, not based on the differentiation (i.e. a local property such as valley), but on the integration (i.e., a global property) of the histogram.", cited in [Otsu, 1979]. The term "**integration**" here implies that the method considers the overall distribution of pixel intensities across the entire image, rather than focusing on specific points or small regions. By integrating the information across the entire histogram, Otsu's method is able to determine a threshold that maximizes the separation between classes globally, which tends to produce more stable and reliable results in various conditions.

In comparison to the naive thresholding method as described in Section 2.1 where we set a threshold based on simple metrics like a fixed value, Otsu's method is more applicable. Although the naive method is computationally efficient, it lacked the adaptability needed for different image conditions. Unlike Otsu's method, which calculates an optimal threshold by analyzing the entire image histogram, the naive method didn't consider the pixel intensity distribution, leading to inferior results in images with varying lighting or noise. Otsu's method, while more adaptive, requires greater computational effort. The following figures shows the difference between the naive thresholding method and the Ostu based segmented image based on Figure1.

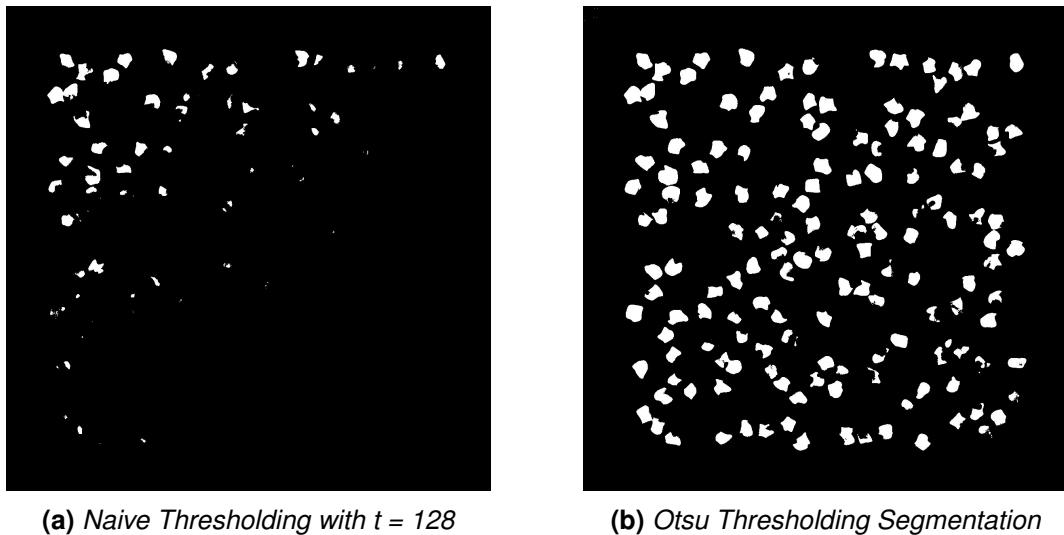


Figure 7 Comparison of Naive vs. Otsu's Thresholding on Cell Image

The key difference between the above sub figures shown in **Figure 7** lies in the effectiveness of segmentation. The Naive Thresholding with $t = 128$ results in incomplete segmentation with many cells either not fully captured or merged into the background. In contrast, Otsu's Thresholding Segmentation produces a more accurate and complete separation of the cells from the background, demonstrating its ability to adapt to varying intensity levels within the image.

Despite its robustness, Otsu's method has several **shortcomings**. One major issue is its sensitivity to noise. Noise can distort the histogram, leading to incorrect thresh-

olding. **Modern approaches** such as adaptive thresholding or local thresholding algorithms, like **Sauvola's method**, address this limitation by computing thresholds for smaller regions of the image [Sauvola and Pietikäinen, 2000]. This method improves upon traditional global thresholding techniques by calculating the threshold for each pixel based on the local mean and standard deviation within a neighborhood around that pixel, being more resilient to noise and lighting variations, particularly in scenarios like document image processing where conditions may vary significantly across the image. Additionally, Otsu's method is inherently designed for bimodal histograms, limiting it to two classes. The method might struggle to accurately segment images with more complex distributions, like multimodal ones. Modern alternatives such as k-means clustering or techniques based on machine learning can manage several classes by examining the image in greater detail and leveraging labeled data to enhance segmentation precision. A modern approach of using **Fully Convolutional Neural Networks** managed to address this issue by introducing a model that could predict labels for individual pixels, leading to improved segmentation in complex images [Long et al., 2015]. These networks were trained to categorize each pixel into various classes, enabling thorough segmentation in complex images. FCNs are very powerful as they can be trained on extensive datasets and adapt effectively to different segmentation tasks, making them well-suited for intricate, multimodal images.

2.4 Edge-detection

Edge detection, an essential stage in image processing, is utilized to recognize the borders of objects in an image. Typically, these boundaries align with regions containing a noticeable change in pixel brightness. Precise edge detection is essential for numerous tasks like computer vision, image segmentation, and object recognition. We applied the Sobel, Scharr, and Prewitt edge detection filters to Figure1 for this task. All of the edge detection filters utilized for this task rely on gradient operators that assess the rate of change in pixel brightness throughout an image. By applying filters to the original image, edges can be detected in either horizontal (X) or vertical (Y) orientations through the calculation of gradients in both directions.

The **Sobel filter** is among the edge detection operators that are frequently utilized. Two 3x3 convolution filters, one detecting horizontal edges (SobelX) and the other vertical edges (SobelY), are applied to calculate the image intensity gradient. The kernels' design emphasizes the regions with high spatial frequency that are associated with edges. This filter operator utilizes dual 3x3 convolution kernels for approximating the gradients in the horizontal (X) and vertical (Y) orientations, as shown below:

$$X = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}, \quad Y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}. \quad (10)$$

where by applying these two kernels in X and Y direction, on an Image (I), as shown in Figure 1, results in two derivatives, named as G_x and G_y . The image gradient is then

calculated using euclidean norm over both derivatives at each pixel in the image:

$$G = \sqrt{G_x^2 + G_y^2}. \quad (11)$$

The **Scharr filter** is a more accurate version of the Sobel filter, designed to enhance gradient magnitude precision and rotational symmetry. This filter is particularly useful when precise edge recognition is needed, such as in photos with complex details or subtle changes in intensity, especially in the detection of diagonal edges. Below are the two 3x3 convolution filters used to calculate the gradients in the horizontal (X) and vertical (Y) directions for this filter.

$$X = \begin{bmatrix} 47 & 0 & -47 \\ 162 & 0 & -162 \\ 47 & 0 & -47 \end{bmatrix}, \quad Y = \begin{bmatrix} 47 & 162 & 47 \\ 0 & 0 & 0 \\ -47 & -162 & -47 \end{bmatrix}. \quad (12)$$

The gradient is calculated in the same manner as Sobel, shown in Equation 11. This filter is known for its higher precision in detecting edges. This is because its kernels are optimized to better approximate the derivative of the image.

Another gradient-based edge detector is the **Prewitt filter**. It also uses two 3x3 convolution kernels to calculate the gradient of the image intensity, just like the Sobel filter, but with a slightly simpler set of convolution kernels. The Prewitt operator is generally less sensitive to noise than Sobel, but it also provides a less precise gradient approximation. The kernels used for this filter are:

$$X = \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}, \quad Y = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}. \quad (13)$$

and the gradient is again calculated in the same manner as Sobel and Scharr, given in Equation 11. Prewitt filters are appropriate for situations where noise reduction is a priority since they are frequently less susceptible to noise than Sobel filters. It does, however, give a less precise approximation of the gradient.

While the Scharr, Prewitt, and Sobel filters are frequently utilized for edge detection, there are certain **limitations** that require the consideration of more sophisticated techniques. One of the primary drawbacks is their vulnerability to noise. These filters are prone to identifying fake edges because they compute the gradient of the image intensity, which is typically caused by small intensity fluctuations brought about by noise in the image [Gonzalez, 2009]. In real-world photos, where noise is inevitable, this problem is especially troublesome. Additionally, these filters are not as effective at detecting diagonal or curved edges since they are designed to primarily detect edges aligned with the horizontal and vertical axes. In images with complicated geometries, this directional bias may result in partial edge detection [Jain et al., 1995]. Furthermore, the fixed size of the convolution kernels restricts their ability to detect edges of

various sizes in an image, potentially causing the loss of important details or the failure to capture bigger structures [Marr and Hildreth, 1980]. Finally, the approximate gradient calculated by these filters often leads to inaccurate edge detection, particularly in areas with sharp gradient [Canny, 1986]. These limitations emphasize the need for more sophisticated edge detection strategies, such the Canny edge detector [Canny, 1986], which combines approaches including edge thinning, gradient magnitude thresholding, and noise reduction to produce more precise and dependable edge detection.

The following figures were generated to show the results of Edge detection after applying the above mentioned filters on Figure 1. The outermost rows and columns were ignored for simplicity.

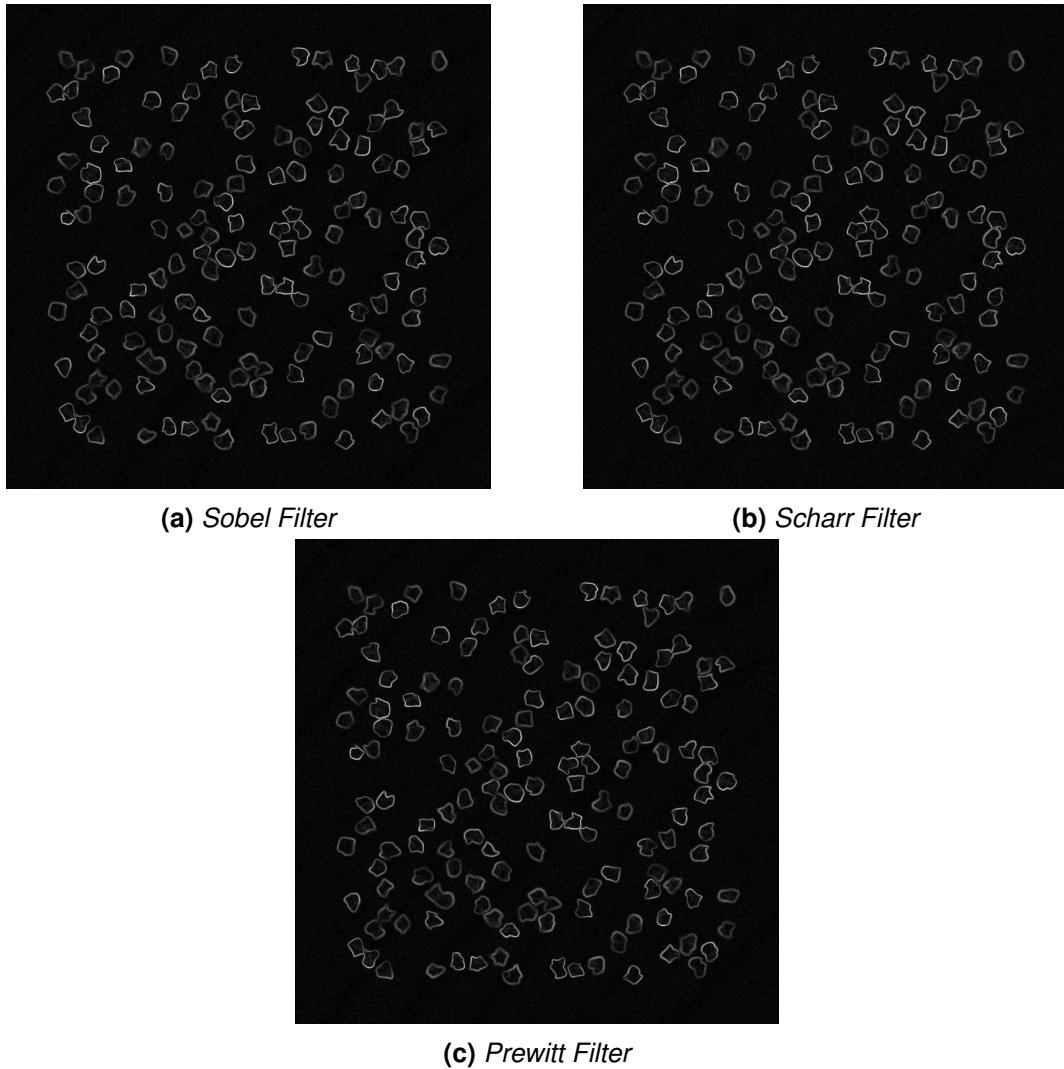


Figure 8 *Results of Primitive Edge Detection Filters*

In conclusion, all three operators — Sobel, Scharr, and Prewitt effectively detect the edges of the cells particles, as shown in Figure 8 given the relatively simple nature of the image. Due to the smoothness of the edges and the consistency in the image's gradient, the differences between these operators are subtle. The Scharr operator, while providing slightly sharper and more pronounced edges, aligns with its design for

detecting finer edge details. However, in this specific image, these differences are not significantly pronounced.

2.5 Canny edge detection

A renowned method in the field of image processing, Canny Edge Detection is renowned for its ability to recognize a wide range of edges in images. John F. Canny developed the method in 1986 with the express purpose of improving edge identification by reducing noise while preserving important structural elements. It achieves this by means of a multi-phase procedure, to maximize edge detection accuracy and dependability [Canny, 1986].

In order to accurately identify edges, the Canny Edge detection algorithm follows a sequence of important steps. In order to decrease the noise that may lead to inaccurate edge detection, an optimal Gaussian blur value is applied to the image for smoothing. Next, the method employs Sobel kernels in convolution to calculate the gradients G_x and G_y of the blurred image along the horizontal and vertical direction, respectively. Potential edges are shown by areas of fast intensity change, which are highlighted by the gradient magnitude $|\nabla G|$. A description of these two steps can also be found in Section 2.4. The arctangent function is utilized for determining the **direction of the gradient** like the following:

$$\theta(x, y) = \text{atan2}(G_y, G_x) \quad (14)$$

where $\theta(x, y)$ represents the gradient direction at each pixel, which is crucial for determining the orientation of the edges during the non-maximum suppression step.

Subsequently, Non-maximum suppression is applied to thin out the detected edges by retaining only the points that are the local maxima in the direction of the gradient, effectively reducing the edges to a single-pixel width. Subsequently, two thresholds T_{low} and T_{high} are used to categorize edges in double thresholding: edges above T_{high} are considered strong, while edges between T_{low} and T_{high} are considered weak. This concept is also known as **Hysteresis Thresholding** which removes unneeded edges while preserving key ones by retaining weak edges only if they connect to strong edges. This method produces a more accurate edge map that accurately represents the image's outlines.

The Canny Edge Detection algorithm offers **significant advantages** over simpler methods like Sobel, Scharr, and Prewitt, particularly in handling noise, improving edge localization, and preserving edge connectivity. The primitive methods, that directly compute image gradients are very prone to noise, leading to frequent generation of incorrect edges. On the other hand, the Canny algorithm starts by applying a Gaussian blur to minimize noise and improve the precision of edge detection. Also, primitive edge detection techniques also result in thicker edges that are less accurately positioned because of basic thresholding. Canny tackles this issue by implementing non-maximum sup-

pression, making sure that edges are both slim and accurately positioned. Additionally, Canny stands out in preserving edge connectivity, a task that simpler techniques frequently find challenging in noisy settings. By implementing double thresholding and hysteresis-based edge tracking, it retains only the crucial, connected edges, resulting in a more dependable and uninterrupted edge detection.

In general, the Canny algorithm outperforms simpler methods by efficiently decreasing noise, precisely identifying edges, and guaranteeing connectivity, making it a more reliable option for edge detection. The parameters selected, notably the Gaussian blur sigma (σ) and the upper and lower thresholds, have a substantial impact on the effectiveness of the Canny Edge Detection algorithm. These parameters determine the algorithm's efficiency to detecting edges and its ability to reduce noise effectively.

Three images are shown below with variations in parameter settings for thresholds and σ value for Gaussian blur, implemented on the same cellular structure image, as seen in Figure 1.

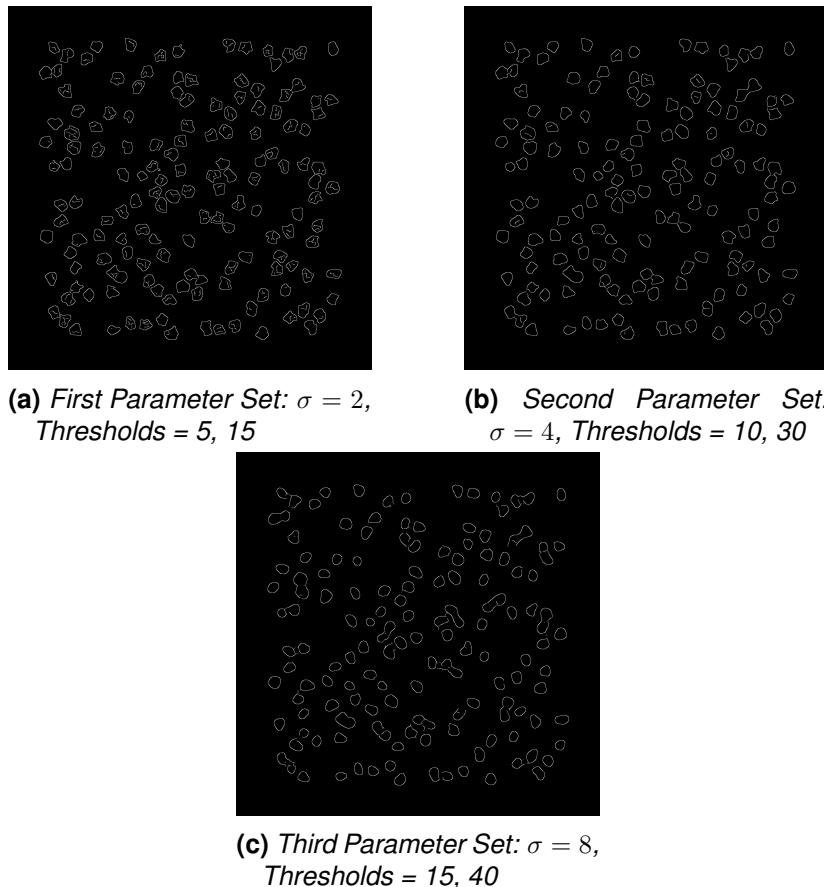


Figure 9 Comparison of Canny Edge Detection results with different parameter sets.

Figure 9a demonstrates that by applying a mild Gaussian blur, noise is decreased while detail is maintained. The thresholds enabled the identification of the majority of edges, leading to an intricate edge map. On the contrary, Figure 9b and Figure 9c illustrates that a higher sigma resulted in more effective smoothing, resulting in decreased noise

while also causing a loss of finer details. Increasing the thresholds resulted in more discerning edge detection, only detecting the most prominent edges.

These findings demonstrate the compromises associated with selecting parameters. Decreasing sigma and thresholds leads to increased precision in detecting edges, detecting even the fainter ones. Conversely, higher values lead to smoother edges yet with less detail. The parameters selection should be based on the particular needs of the image analysis task.

3 Conclusion

By creating fundamental image processing tools like Thresholding, Segmentation, and Edge detection techniques, this project has established the foundation for understanding and utilizing advanced image analysis methods. In the field of medical imaging, segmentation and edge detection are extremely important methods and should not be underestimated. These techniques play a key role in automating tasks like accurately identifying anatomical features and abnormal changes. Their use could greatly help radiologists by improving both efficiency and accuracy. Now, we will conclude this report with the current trends being used in the field of image processing. Now, we will conclude this report with the current trends being used in the field of image processing

Currently, deep learning methods, particularly **Convolutional Neural Networks** (CNN), are revolutionizing image segmentation, especially in medical applications such as organ segmentation and tumor detection. The U-Net architecture, for example, has demonstrated outstanding performance in these areas [Ronneberger et al., 2015]. Additionally, **Multi-Modal Image Fusion**, which combines data from different imaging modalities like MRI and PET, is gaining attention for its potential to enhance diagnostic precision, though it faces challenges in clinical implementation due to the complexity of data integration and computational demands [James and Dasarathy, 2014].

Transfer learning is also emerging as a solution to the challenge of limited labeled data in medical imaging. By adapting pre-trained models to specific medical tasks, it enables the rapid deployment of effective models with reduced training times, though it requires further validation across diverse clinical settings [Cheplygina et al., 2019].

Among these trends, deep learning segmentation using CNNs is the most advanced and is increasingly being incorporated into clinical practice, particularly for tumor identification and organ delineation, but still the issues for handling large amounts of training data and having enough computational power exist. In contrast, transfer learning and multi-modal image fusion remain primarily in the research stage, with their clinical application contingent on overcoming validation, regulatory, and integration challenges. Transfer learning shows potential for situations with a lack of data, but it must undergo thorough validation in various clinical settings to guarantee its dependability and applicability before widespread adoption. Likewise, the process of combining data from multiple imaging modalities in multi-modal image fusion is inhibited by the complexity and computational requirements

The report effectively showed various basic image processing techniques such as thresholding, segmentation, and edge detection, as they were successful in solving key image analysis challenges. Nevertheless, some limitations are there that could affect the overall effectiveness of these methods. One main issue is that thresholding and edge detection methods are sensitive to changes in image quality and lighting conditions. Even though histogram equalization was used to address lighting differences, the effectiveness of these techniques varied among images, leading to less than ideal results for segmentation and edge detection. Additionally, the reliance on manual parameter adjustment in techniques such as Otsu's thresholding and the Canny edge

detector can lead to possible user bias and result variability. This manual intervention not just decreases reliability but also limits the scalability of these techniques when used on extensive datasets or in clinical environments. Furthermore, though these techniques are usually successful in analyzing relatively uncomplicated images, their use for more detailed medical images, which frequently contain elaborate structures, is still limited.

To address these limitations, future studies should concentrate on enhancing the dependability of these methods, particularly across various imaging settings. This could involve integrating adaptive algorithms or using machine learning methods to automatically modify parameters and enhance the management of different imaging scenarios. As research advances, these developments may transition from experimental to common clinical practice, resulting in significant improvements in diagnostic imaging and leading to more accurate and efficient medical assessments.

References

- [Canny, 1986] Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, (6):679–698. 2.4, 2.5
- [Cheplygina et al., 2019] Cheplygina, V., De Bruijne, M., and Pluim, J. P. (2019). Not-so-supervised: a survey of semi-supervised, multi-instance, and transfer learning in medical image analysis. *Medical image analysis*, 54:280–296. 3
- [Gonzalez, 2009] Gonzalez, R. C. (2009). *Digital image processing*. Pearson education india. 2.1, 2.4
- [Huang and Wang, 1995] Huang, L.-K. and Wang, M.-J. J. (1995). Image thresholding by minimizing the measures of fuzziness. *Pattern recognition*, 28(1):41–51. 2.3
- [Jain et al., 1995] Jain, R., Kasturi, R., Schunck, B. G., et al. (1995). *Machine vision*, volume 5. McGraw-hill New York. 2.4
- [James and Dasarathy, 2014] James, A. P. and Dasarathy, B. V. (2014). Medical image fusion: A survey of the state of the art. *Information fusion*, 19:4–19. 3
- [Long et al., 2015] Long, J., Shelhamer, E., and Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440. 2.3
- [Marr and Hildreth, 1980] Marr, D. and Hildreth, E. (1980). Theory of edge detection. *Proceedings of the Royal Society of London. Series B. Biological Sciences*, 207(1167):187–217. 2.4
- [OpenAI, 2024] OpenAI (2024). Chatgpt: Gpt-4. <https://www.openai.com>. Accessed: 2024-08-24. 1
- [Otsu, 1979] Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66. 2.3, 2.3
- [Ronneberger et al., 2015] Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5–9, 2015, proceedings, part III 18*, pages 234–241. Springer. 3
- [Sahoo et al., 1988] Sahoo, P. K., Soltani, S., and Wong, A. K. (1988). A survey of thresholding techniques. *Computer vision, graphics, and image processing*, 41(2):233–260. 2.1
- [Sauvola and Pietikäinen, 2000] Sauvola, J. and Pietikäinen, M. (2000). Adaptive document image binarization. *Pattern recognition*, 33(2):225–236. 2.3

- [Setiawan, 2020] Setiawan, A. W. (2020). Image segmentation metrics in skin lesion: accuracy, sensitivity, specificity, dice coefficient, jaccard index, and matthews correlation coefficient. In *2020 International Conference on Computer Engineering, Network, and Intelligent Multimedia (CENIM)*, pages 97–102. IEEE. 2.2
- [Sezgin and Sankur, 2004] Sezgin, M. and Sankur, B. I. (2004). Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic imaging*, 13(1):146–168. 2.3
- [Smith et al., 2015] Smith, K., Li, Y., Piccinini, F., Csucs, G., Balazs, C., Bevilacqua, A., and Horvath, P. (2015). Cidre: an illumination-correction method for optical microscopy. *Nature methods*, 12(5):404–406. 2.1
- [Van Stralen et al., 2009] Van Stralen, K. J., Stel, V. S., Reitsma, J. B., Dekker, F. W., Zoccali, C., and Jager, K. J. (2009). Diagnostic methods i: sensitivity, specificity, and other measures of accuracy. *Kidney international*, 75(12):1257–1263. 2.2
- [Wang et al., 2021] Wang, J., Wang, X., Zhang, P., Xie, S., Fu, S., Li, Y., and Han, H. (2021). Correction of uneven illumination in color microscopic image based on fully convolutional network. *Optics express*, 29(18):28503–28520. 2.1