**Bacterial Colony Counter using Different Image**

**Processing Algorithms**

by

**Kenji Kevin Fulcher**

**Ivan Gil G. Mercano**

**Eric Miguel Parma**

A Thesis Paper Submitted to the School of Graduate Studies in Partial Fulfilment of the Requirements for the Degree

Bachelor of Science in Computer Science

Mapúa University

July 28, 2022

**Table of Contents**

TITLE PAGE ................................................................................................................................................ i

TABLE OF CONTENT ............................................................................................................................... ii

[LIST OF TABLES iii](#_Toc115196004)

[LIST OF FIGURES iv](#_Toc115196005)

Chapter 1: [INTRODUCTION 4](#_Toc115196006)

Chapter 2: [REVIEW OF RELATED LITERATURE 9](#_Toc115196007)

Chapyer 3: [METHODOLOGY 23](#_Toc115196008)

[Methodology 23](#_Toc115196009)

[Conceptual Framework 24](#_Toc115196010)

[Data Acquisition 25](#_Toc115196011)

[Dataset 26](#_Toc115196012)

[Dataset Annotated Values 27](#_Toc115196013)

[Dataset Thresholding 28](#_Toc115196014)

[Grayscale Conversion and Petri Dish Bounds Detection 28](#_Toc115196015)

[Initial Procedures on Detecting Colonies 29](#_Toc115196016)

[Binary Conversion 29](#_Toc115196017)

[Morphological Processing 29](#_Toc115196018)

[Colony Edge Filtering 30](#_Toc115196019)

[Circle Detection 32](#_Toc115196020)

[Alternative Procedures on Detecting Colonies 34](#_Toc115196021)

[Contrast Improvement 34](#_Toc115196022)

[Denoising 36](#_Toc115196023)

[Blob Detection 37](#_Toc115196024)

[Colony Counting 38](#_Toc115196025)

[Testing 38](#_Toc115196026)

Chapter 4: [RESULTS AND DISCUSSION 39](#_Toc115196027)

[Kis Implementation of Circular Hough Transform 39](#_Toc115196028)

Chapter 5: [CONCLUSIONS AND RECOMMENDATIONS 65](#_Toc115196029)

[Conclusions 65](#_Toc115196030)

[Recommendations 66](#_Toc115196031)

[References 100](#_Toc115196032)

# LIST OF TABLES

**Table 1.1** Analysis of Image Processing ..… Colony Detection and Counting ………….…......………. 20

**Table 1.2** Algorithm Thresholding ………………………………………………….…………………... 23

**Table 2.1** Machine Specification …………………………………………………….………..………… 24 **Table 4.1** Range of Parameters for the Circular Hough Transform using the Study of Kis et al. ……..... 46

**Table 4.2** Range of Parameters for the Circular Hough Transform using the Alternative Process …….. 49

**Table 4.3** Range of Parameters for the Blob Detection Algorithm using the Alternative Process …...… 49

**Table 4.4** Opaque Colony Count Results for the Process of Kis et al ………………………………….. 53

**Table 4.5** Opaque Colony Counts for the Alternative Process using Circular Hough Transform ............ 54

**Table 4.6** Opaque Colony Count for the Alternative Process using Blob Detection ………...…........…. 55

**Table 4.7** Independent Sample T-Test Result for Opaque Colonies Only ……………………….……... 56

**Table 4.8** Translucent Colony Count Results ….. using Circular Hough Transform …………………... 56

**Table 4.9** Translucent Colony Count for the Alternative Process using Blob Detection …...................... 57

**Table 4.10** Independent Sample T-Test Result for Translucent Colonies Only………............……….... 58

**Table 4.11** Combined Colony Count Results for the Method of Kis et al. ……………………………... 59

**Table 4.12** Combined Colony Count Results ….. using Circular Hough Transform ............................... 60

**Table 4.13** Combined Colony Count for the Alternative Process using Blob Detection ……………….. 60

**Table 4.14** Independent Sample T-Test Results for Both Opaque and Translucent Colonies .................. 61

**Table 4.15** Average Performance Overview ……………………………………………………………. 62

# LIST OF FIGURES

**Figure 1.1** Conceptual Framework ........................................................................................................... 26

**Figure 1.2** Data Acquisition Process Flow................................................................................................ 27

**Figure 1.3** Sample Containing Translucent Colonies (Marked Manually)............................................... 28

**Figure 1.4** Samples Categorized as Dark (Left), Bright (Middle), and Vague (Right)............................. 28

**Figure 1.5** Grayscale Image and Petri Dish Bounds Detection................................................................. 29

**Figure 1.6** Binary Image (left), Erosion (center), Dilation (right)............................................................. 31

**Figure 1.7** Visualization of the Sobel Operator Convolution and Hysteresis Thresholding .................... 33

**Figure 1.8** Circular Hough Transform using Three Edge Pixels ............................................................. 34

**Figure 1.9** Results of the Circular Hough Transform drawn on a Sample Image .................................... 34

**Figure 1.10** Example of a Cumulative Frequency Distribution................................................................. 35

**Figure 1.11** Effect of Histogram Equalization on Frequency Distribution............................................... 36

**Figure 1.12** Histogram Equalization Visualization................................................................................... 36

**Figure 1.13** Non-Local Means Denoising................................................................................................. 38

**Figure 1.14** Blob Detection Results.......................................................................................................... 39

**Figure 1.15** T-table.................................................................................................................................... 42

**Figure 4.1** Grayscale Image ………………………….………………………………………….……… 43

**Figure 4.2** Petri Dish Bounds Detection ................................................................................................... 43

**Figure 4.3** Translucent (Red) and Opaque (Green) Bacterial Colony............................................... 44

**Figure 4.4** Binarized Image........................................................................................................................ 45

**Figure 4.5** Denoised Image........................................................................................................................ 45

**Figure 4.6** Colony Detection on a Binary Image using the Circular Hough Transform ........................... 47

**Figure 4.7** Histogram Equalization ........................................................................................................... 48

**Figure 4.8** Non-Local Means Denoising ................................................................................................... 48

**Figure 4.9** Opaque Colony Detection Using Circular Hough Transform.................................................. 50

**Figure 4.10** Translucent Colony Detection Using Circular Hough Transform ............................... 51

**Figure 4.11** Opaque Colony Detection Using Blob Detection Algorithm ................................................ 52

**Figure 4.12** Translucent Colony Detection Using Blob Detection Algorithm ......................................... 53

**Figure 4.13** False Positives caused by the Circular Hough Transform (Translucent) .............................. 63

**Figure 4.14** Overlapping Colonies Treated as One by Blob Detection (Translucent) ............................. 64

**Figure 4.15** Overlapping Colonies that are not Detected by Blob Detection (Opaque) ........................... 65

**Figure 4.16** False Positives caused by Histogram Equalization and Blob Detection (Translucent) ........ 66

**Figure 4.17** Letter ‘S’ Detected by Blob Detection .................................................................................. 67

**Figure 4.18** False Negatives Caused by Shadows Near the Edges............................................................. 68

**Chapter 1**

# INTRODUCTION

During the past few decades, technology has advanced rapidly and become an essential factor in our society. Computers can now acquire and process raw information from their environment and apply the processed data to solve or answer a given question or problem. One of the most vastly applicable computer processes right now is computer vision. Computer vision is an Artificial Intelligence (AI) field where the computer derives data from visual inputs such as images and videos. The computer is then used the derived data to take action and recommend a possible approach to solve or obtain the main objective [1]. Unlike humans, a computer processes the data with the help of a designed algorithm to obtain the specific information from the given visual data.

Typically, computer vision is used in areas wherein there is a repetitive task of manual inspection in order to alleviate its time consumption. It has a wide domain that can cover security, automobiles, manufacturing, agriculture, and microbiology among others.

In the security domain, computer vision took part in tracing, identifying potential hazards that may be present in both public areas and workplaces. It provides insights that can prevent particular accidents from happening by collecting visual information from its environment. One of the existing applications of computer vision in security is an AI model called CHOOCH. This model [2] provides tracking and monitoring of safety health guidelines in the workplace and for the public, facial authentication, remote perimeter sensing, electro, and infrared object detection tool.

In automobiles, computer vision is mostly applied in the automation of manufacturing processes. By computer vision, [3] automobile industries efficiently enhance the company’s production process and reduce the manual labor required in the process by using robotic guidance that can set given parameters to accurately do a job and inspection to maintain the quality of each job done in the process of manufacturing vehicles.

Another field covered by computer vision is agriculture. Agricultural industries stepped into computer processes in order to improve the product in crop cycles. With the help of advanced visual technology such as drones and satellites, gathering data for a large space becomes possible. Different agricultural applications were applied such as plant health detection, weather condition analysis, harvesting, and planting, etc. One example of an existing computer vision application in agriculture that exists today is Drone-based crop monitoring [4] where a drone is used in surveying the crop field to detect unwanted conditions by analyzing crop health information and soil conditions, weather analysis, and monitoring the crop fields.

Computer vision applications are also present in microbiology; in particular—the counting of bacterial colonies within a petri dish. Bacteria or prokaryotes are microscopic single-celled organisms with a simpler cell structure than other organisms. It lacks other membrane-bound organelles such as the nucleus. It is only composed of deoxyribonucleic acid (DNA), chromosomes, and ribosomes inside the plasma membrane protected by the cell wall and bacterial capsule. Unlike most other organisms, bacteria reproduce by binary fission; it is a process where an organism grows and splits its body along with its genetic material to become two separate organisms [5]. Originally, we cannot see bacteria with our naked eye, because of this reason, scientists purposely cultured bacteria in order for them to form bacterial colonies and become observable.

A bacterial colony is commonly known as a group of bacteria that originated from the same host cell. Meaning that the host cell is responsible for creating genetically identical groups of bacteria. Colony counting is the process of identifying colonies within a bacterium on an agar plate, this process could either be done manually or automatically. Manually, the researchers would have to first identify the bacteria and decide if it is a colony, once the bacteria meet the minimum criteria of a colony, the tally increases, and the researchers move on to the next sample. Automatically, the machine has predetermined thresholds on identifying if the sample is a colony or not, the speed and accuracy outmatch that of the manual method.

Several advances in the usage of computer vision in counting circular bacterial colonies have been made in recent years. Hogekamp et al. [6] extract the local maxima of each colony in order to count them, this refers to extracting the darkest pixel within a group of pixels and using that in order to count the colonies. This, however, used fixed parameters and required tuning whenever a new set of samples was introduced. In addition, this method performed poorly against samples with translucent colonies since the brightness level across each colony is almost uniform. García-Sorano et al. [7] used k-means clustering for both static images and videos in order to monitor changes within colonies over time. This method instead counts by consolidating pixels whose colors are close to each other to an average color to meet a set number of classes. However, it required that illumination and exposure in the images were consistent across all the samples. Beznik et al. [8] on the other hand, made use of convolutional neural networks which were supervised by labeling areas of the image as either background or colonies but performed poorly when faced with colliding colonies, having a precision and recall score of 50.3% and 28.5% for the border class, respectively.

Kiş et al. [9] used the Circular Hough Transform, which made use of the equation of a circle in order to count the colonies, and achieved great results with error rates as low as 1.14% for E. coli colonies which happened to be easily contrasted from the background by an untrained eye from the start, but lacked some preprocessing which resulted in translucent colonies being ignored, increasing the error rate to 10.92% for E. faecalis colonies, which appeared to be translucent in the image. In addition, the Circular Hough Transform becomes unreliable when the circles to be detected have blurry edges, of which must be detected as a prerequisite; and have a large range of radii. The combination of these two scenarios means that the minimum value of the accumulator must be set lower to detect these types of colonies but doing so will also increase the possibility of creating false positives.

Since colonies can also appear as blobs or Binary Large Objects, blob detection could be used in detecting them. Grossi et al. [29] explains that blob detection groups together adjacent pixels of the same color in a binarized image and has the advantage of filtering blobs based on properties such as area, circularity, inertia, and convexity, unlike Circular Hough Transform which is limited to perfect circles. Loddo et al. [30] used this algorithm to detect white blood cells or leukocytes from microscopic images and classifies them to detect leukemia using a convolutional neural network, boasting a 99.7% detection accuracy. Like that of the aforementioned studies, detection of translucent colonies has yet to be improved since blob detection relies on blobs which persist across many threshold levels of binarization.

Despite several differences in counting techniques, the trend found in previous research indeed appears to be the reduced performance in situations wherein some extra preprocessing can help improve the results. In the studies conducted by Hogekamp et al. [6], Kis et al. [9], Grossi et al. [29], and Loddo et al. [30], this pitfall occurs when the colonies appear translucent, causing a lack of contrast in color intensity or brightness between them, of which thresholding and segmentation heavily rely on. Thus, this study aims to answer the question:

* How will a bacteria colony detection model perform if the image processing techniques were altered to compensate for the algorithm' deficiency in detecting colonies that have translucent characteristics?

Based on this problem, the main objective of the study thereafter is:

* To propose and determine set of combination of algorithms to preprocess the images and reliably detect bacterial colonies that have translucent characteristics.

The importance of solving the problem is to achieve an alternative method of counting the bacterial colonies and to acquire accurate and consistent data from the samples using image processing and feature extraction. The beneficiaries of the study would be:

1. **Current Researchers** - The current researchers will be able to expand the field and improve already existing algorithms by combining or adding other components that can improve the algorithm in classifying objects in computer vision.
2. **Experts in The Medical Field** - Experts in the medical field will be able to have an alternative method of counting bacterial colonies for additional validation and consistency.
3. **Patients** - The diagnosis and procedures for the patient would be improved as well.
4. **Future Researchers** - Future researchers will have a point of reference in expanding the related fields of the research.

The sample images to be used will be acquired from the AGAR dataset, which contains roughly 18000 images of microbial colony samples that are categorized as bright, dark, or vague. A consolidated dataset consisting of only 30 images that are under the ‘vague’ category, which consists of samples with translucent colonies, will be used. The opaqueness of the colonies is a critical factor in the selection of the samples since these are the types of images that the previous studies struggled to perform. The classification of bacteria is not important, as the objective of this study is to obtain the count only.

**Chapter 2**

# REVIEW OF RELATED LITERATURE

A paper by Hogekamp et al. [6] focuses on the development of an image analysis process for counting bacteria colonies using an ImageJ macro. Only colonies of E.coli DH5a are used and the parameters used in the macro are not dynamic, meaning that a specific setup was needed for this method to work. Particularly, regions of interest (ROIs) are not automatically detected which meant that the petri dish must be placed in a fixed position in the image. This macro detects colonies by finding the local maxima of each colony and claims that it has better recognition of overlapping colonies than thresholding and segmentation algorithms. Counting rates are found to be 1.9 colonies per second manually (on the petri dish), 1.1 colonies per second using the gold standard method (manually counting within the program), and a fixed 20 seconds for any number of colonies using the automatic method. Compared against the gold standard at 95% confidence, manual counting achieved a confidence interval of (0.896, 0.93) and automatic counting achieved (0.972, 0.98). This research recommends that a comparison between the original image and the resulting image after determining its maxima must be done for plates with tightly spaced colonies by stacking the images on top of each other.

An article written by Kis et al. [9] aims to count the number of colonies of bacteria present in a petri dish using a GUI application with the Circular Hough Transform, created in MATLAB. The images needed to be captured in a black box and only circular-shaped colonies were used, specifically Escherichia coli, pseudomonas aeruginosa, and enterococcus faecalis, and its error rates were 1.14%, 1.69%, and 10.92%, respectively. Enterococcus faecalis has a higher error rate since it is not as opaque in the microscopic image as the other two colonies. They suggested that the program can be improved using the data of different morphologically structured colonies; an extension of the program that can monitor changes in colonies in real-time can be implemented in the future.

A paper by Garcia-Soriano et. al [7] aims for the development of an open-access tool - ColFeatures, for the identification of bacterial colonies using different machine learning algorithms and the tracking of its morphological features such as the location of its centroid, its area, diameter, perimeter, circularity, eccentricity, and convex hull over time. This tool is able to monitor the changes within colonies over time. The bacteria samples were sourced from soil found in Mols Berge National Park, Denmark. This tool makes use of k-means clustering, where ‘k’ is determined by the Silhouette Coefficient, Calisnki-Harabasz score, and Davies-Bouldin score. They claim that this tool is able to detect morphological features that cannot be seen by an untrained eye. However, improvements can be made—using dark agar plates are less effective, pictures must have high pixel quality, illumination and exposure time must be consistent, images must stay in the same position when recording changes over time, and fused bacteria identification is sacrificed to retain morphological information.

Similarly, Marquard et al. [10] wrote a paper that aims to create an image analyzing app for handheld devices that can detect E.coli colonies on agar plates using Digital Analyzer, an inhouse software written in C#. Like Kis et al. [9] the main algorithm used to count colonies is the Circular Hough Transform. However, the circles to be detected were defined by a radius that is ranging from the threshold that is one-third of the average object size (AOS) to itself. The mean difference between manual counting and automatic counting was 4.12%, however, it may change depending on the density of colonies on the plate. Low densities had a mean difference of 4.32% and high densities had a mean difference of 3.70%. The recognition performance in the rim area of the plate could be improved in the future without relying on a correction factor, and the program should be implemented into an actual mobile application.

A study conducted by Beznik et al. [8] focuses on developing a U-Net CNN that can detect bacterial colonies and classify them as virulent or avirulent. 108 images of petri dishes containing bacterial colonies were used in this research. The CNN was able to achieve greater than 90% precision and recall for the virulent and and avirulent classes; however, the border class, which determines if two colonies are colliding, only achieved around 50% precision and 30% recall. The CNN is not able to completely separate two colliding colonies despite detecting some border pixels. The researchers suggested that a larger dataset must be explored.

A paper written by Huang and Tong [11] aims to create an automatic program that can classify a type of bacterial colony from an image using a traditional CNN, AlexNet, and an unsupervised method called Autoencoder. The 18 most common bacteria colonies found in Peking University First Hospital were used in this study. The overall accuracy can reach 73% and individually the classes for all bacteria can reach accuracy and specificity of 90%. CNN and AlexNet both have higher accuracies as the ratio between training data and test data increases; Autoencoder does not have any significant difference in performance as it increases. Finer hyperparameter tuning could be used to improve these results.

A paper written by Talo [12] focuses on developing an automatic process in recognizing and classifying bacteria from a microscopic image. It makes use of Deep Transfer Learning on the ResNet-50 pre-trained convolutional neural network, which reduces the training time by using an already trained and similar CNN to make use of fewer samples. Training and test data is sourced from the DIBaS bacteria species dataset which is composed of 33 different bacterial species. The CNN achieved an accuracy of 99.2%, 99% precision and recall, as well as a 99% F-score. This research claims that this model is ready to be used in clinical microbiology applications.

Stolze et al. [13] conducted a study that aims to detect colonies of Candida albicans within marijuana flower samples using a manually selected region of interest followed by generic thresholding and watershed algorithms in ImageJ. Out of the 15 samples, only 3 results from the analysis achieved p values less than 0.05 which meant that the manual and automated count were statistically different, however they were found to be caused only by user error. The research recommends using the program as electronic evidence when determining if a sample has less than 104 colony-forming units.

Altuntas et. al [14] measured the performance of different segmentation methods when counting the number of molds, bacteria, or yeast colonies found in a petri dish. The methods compared were Otsu thresholding, Hough transform, Gabor segmentation, watershed segmentation, and k-means clustering. The colonies used originated from dairies found in Marmara, Turkey. The research found that less noisy images are produced using daylight and a 3point angular led lamp with a black background. Despite this, Otsu thresholding is not effective for bacteria and yeasts due to the medium in which the image is produced wherein the image suffers some lack of contrast between the two. Gabor segmentation suffers when different colors and textures are present. Colonies can merge or split depending on the parameters of the k-means clustering algorithm. Overall, the watershed algorithm achieved the highest TCCR+PCOR score of 1.72 but also the highest NIR of 4.65. Thus, it is recommended that analyses produced by the watershed algorithm must be manually assessed to ensure its precision due to its high NIR. Feature extraction can be applied to classify segmented images.

Andreini et al. [15] conducted a study applying deep learning in segmenting images of a bacterial colony. The researchers introduced a new approach in segmenting bacterial colonies by devising an engine that generates synthetic plate images. The synthetic plates were done by applying streaking simulation to an empty plate; then, it is randomly blended onto the colony dataset. The random colony dataset and synthetic plate dataset were used to train the Fully Convolutional Network algorithm; this network is picked based on its performance in separating the background apart from the bacterial colony in the images. Finally, the researchers concluded that adding the synthetic images in training the Fully Convolutional Network effectively improves the performance of segmenting the images.

Shi et al. [16] developed a noise-free bacterial colony counter by developing a model to identify the noise. The researchers first selected food fragments that can be considered noise in bacterial colony counting. These food fragments are sausage, bacon, and millet. The researchers then extracted the features corresponding to colony clusters and background from agar medium and food fragments. A cluster segmenting model was developed to identify the colony cluster from the background region. The spectral features of the center and border of the colony were extracted, and the colony-separating cluster model was developed. Lastly, each pixel from the agar plate was identified using an established calibration mode; this enabled the bacterial colony on the agar plate to be counted. The researchers concluded that the model could identify the noises or the food fragments with similar color and shape to the bacterial colony.

Wong et al. [17] developed a smartphone based APD colony counter app that processes images of agar plates and automatically counts the bacterial colony in the bacterial culture. Traditionally, bacterial colony counting is manually done and requires a lot of time, patience, and human resources to complete one bacterial culture. The researchers aim to use feasible resources such as smartphones to lessen the vigorous burden that traditional process has. The researchers used watershed algorithm and distance transform for the segmentation process to cluster the colonies effectively. The researchers concluded that the APD colony counter app is efficient in segregating merged colonies better. The application enables smartphone devices to be a helpful tool in the laboratory and becomes an adequate substitute for the commercial bacterial colony counter, which is more costly.

Lin et al. [18] devised a bacterial colony counter on agar plates by applying deep learning techniques. The researchers acquired a total of 252 image samples of bacterial culture in Petri dishes. The researchers developed a fully convolutional neural network in predicting the number of colonies. The central architecture used is FCN from U-Net, which has four down samplings by max pooling and four up sampling. In training the algorithm, the researchers used the loss function by combining the least-squares of error and similar loss. The researchers concluded that the GAN based model using U-Net yields an average error of 7.9 colonies per image. The deep learning approach proves to be efficient in bacterial colony counting if the shape, color, and feature colonies are unknown.

Zhu et al. [19] developed an automated bacterial colony counter that is robust to light. The researchers aim to decrease the error margin caused by light illumination and the reflection of visible light in image samples—the researchers' bacterial cultures on raw cow's milk as identification objects. The methods were done first by pre-processing the images, i.e., eliminating noises, removing the plate edge, identifying and separating overlapped colonies, and counting and labeling colonies using distance transform and watershed algorithm. The method used yielded a relative error of -7.4% ~ +8.3%, with an average relative error of 0.2%. The time used in counting colonies on each plate was 11 to 21s, which is 15 to 75% less than the time consumed in manual counting. The researchers concluded that the developed system in the automatic counting method has an efficient performance in terms of precision and is effective even in certain lighting conditions.

Badieyan et. al [20] aims to evaluate the detection and discrimination in bacterial colonies, to do this the group of researchers studied the polarization properties of various bacterial colonies. The mueller matrix images of four kinds of bacteria were used as the sample images, and the methods used were the mueller matrix polar decomposition, frequency distribution histogram, and central moment analysis method. The result showed that the acquired parameters of the methods were able to provide a quantitative score to be capable of distinguishing different bacterial colonies. It was recommended that more extensive structural studies and modelling of additional kinds of bacterial colony samples were needed to further pursue the research.

Wang et. al [21] aims to create a system capable of detecting live bacteria colonies that periodically captures the growth inside of a 60-millimeter diameter agar plate which will then be analyzed using deep neural networks for detection of the bacterial growth. The study used 3 types of bacteria, which are K. aerogenes, K. pneumoniae, E. coli. The total sample of bacterial colonies used was 965 colonies from 15 petri dishes. The sample bacteria were placed in water as a reagent for the colonies, shortening the detection time. The results showed that their system was able to detect 80% of the colonies within 6 h of incubation for the K. pneumoniae, 6.8 h of incubation for the E. coli and 8.8 h of incubation for the K. aerogenes. Additionally, 90% of the colonies were detected if an additional 1 h of incubation was added, and 95% of the colonies for all three bacterial colonies were detected after 12 h of incubation.

Rao et. al [22] aims to find a solution on counting transformed bacterial colonies from nontransformed ones placed on agar plates, the proposed solution consists of an automated system that is capable of efficiently counting, easy access, and low-cost maintenance. The used methodology was image masking, this involves denoising, edge detection, contour detection, and computation of the convex hall. This technique can detect the amount of blue and white colored colonies post bacterial transformation which is used to calculate the transformation efficiency of the bacteria. The researchers concluded that the automated system is significantly faster than manual counting, though the researchers state that there is a limitation to the system which can be further improved. The problem is that if the background color of the agar plate is changed, additional training and changes are needed for the model to adapt, if future researchers would like to take on a similar study as this, this limitation should be kept in mind.

Lee et. al [23] this study aims to build a system capable of detecting biofilms, specifically Escherichia coli and Salmonella typhimurium on the utensils of food processing facilities, through the usage of fluorescence hyperspectral imaging. The samples were chosen for a specific range of 420-730 nm by emitting UV light from a light source with a 365 nm UV light source. The methodologies used for processing the samples were linear discriminant, k-nearest neighbor, partial-least squares discriminant analysis. The results showed that the sensitivity and specificity of E. coli and Salmonella was over 90% for most of the analysis of the used machine learning models. The researchers concluded that the results from the machine learning models, and the detection of biofilms were well performed. Based on the results, the researchers recommend the idea of integrating biofilm inspections using fluorescence hyperspectral images in other systems.

Treebupachatsakul et. al [24] aims to use image classification and deep learning methods on processing sample images in classifying bacteria. The methodology used by the researchers was done with Python programming and the Keras API. The results showed that most of the sample images of the bacteria were recognized as a type of bacteria by the system. The researchers proposed that for future versions of the study, to use images with higher resolution to improve the accuracy of the tests.

Hamdani et al. [26] developed a method for counting bacterial colonies with low quality images as its samples using image filters and image morphology operators. The image is converted into grayscale before pre-processing were performed. Perona-Malik Diffusion and Contrast Limited Adaptive Histogram Equalization is done to enhance the image. Morphology operators were then used in the petri dish extraction and the colony counting processes.

In search of a contrast increasing algorithm, Byun et al. [27] used histogram equalization to replot the gray levels of each pixel in an image across the entire intensity scale using a cumulative distribution function. This has the effect of highlighting objects in areas wherein there is little contrast to make out the edges of an object. Although this paper focused on cell nuclei detection, its appearance is similar to that of bacteria colonies in shape. In addition, they reported that this algorithm also increases the intensity difference that is associated with noise, thereby increasing its visibility as well. Thus, they recommend using some denoising algorithm to counteract this effect.

A known method of combating noise is done by using the non-local means denoising algorithm. Yang et al. [28] uses this algorithm in denoising live-cell images for particle detection, which also have a similar appearance to bacteria colonies. Unlike regular denoising algorithms which take the mean intensity within one kernel, non-local means denoising takes the mean intensity of multiple patches which are the same size and are similar in terms of intensity values per pixel to the kernel through the mean squared error.

Grossi et al. [29] used blob detection to detect microbial growth of different bacterial colonies. This was done by grouping together adjacent pixels that have the same intensity in a binary image. The Circular Hough Transform was also used, but they found that the blob detection algorithm yielded better results because colonies are often not perfectly circular. Similarly, Loddo et al. [30] also used blob detection for white blood cells in blood sample images which were then fed through a deep learning process to detect leukemia. Prior to using blob detection, they defined a threshold on the hue channel to remove platelets and red blood cells from the image since they have a different color.

Similarly, Loddo et al. [30] also proposed the usage of blob detection in detecting white blood cells from microscopic blood images and then detecting leukemia via a deep learning process. Prior to blob detection, a defined threshold is applied on the hue channel to remove platelets and red blood cells. The system is then tested on public dataset for leukemia detection to test the model’s performance. On the SMC-IDB and IUMS-IDB dataset, the model achieved an accuracy score of 99.7% accuracy, and a 94.1% on the ALL-IDB dataset. Since the white blood cells have a distinct color that differentiates them from the background of the sample images, issues regarding translucency were circumvented.

Zhang et al. [31] developed a novel detector to identify small blobs such as cells or objects that are similar in size. The researchers proposed a Hessian-based Laplacian of Gaussian while having the scale space theory as the foundation of the model. Acquiring the dataset, the researchers implemented a image smoothing algorithm via LoG, then Hessian analysis is launched to pre-determined the scale on which a segmentation will be based. The segmented images were then fitted into an unsupervised clustering algorithm, the clustering will be more robust and sensitive than most of the traditional thresholding algorithm which is commonly used in image detection. To evaluate the performance of the model, the researchers prepared a two set of 2D grey scaled medical images, the datasets were then fitted to the model and compared the results to the results of state-of-the-art medical image detection tool using precision, recall, and F-score metrics. The researchers concluded that the model outperformed the HLog image detection tool on both datasets, but only on par with the GLog and Radial-Symmetry.

Chang et al. [32] developed a computer-aided detection (CADe) system based on multi-scale blob detection for analyzing automated whole breast ultrasound (ABUS). The databased used for the developed model is composed of 136 breast lesion (58 benign lesions and 78 malignant lesions) and 37 normal cases. For the pre-processing, a speckle noise reduction is applied, then the blob detection algorithm was used to identify tumors. Although the model detected the tumors, it was still too sensitive and needed further improvement, to compensate for the lapses, a logistic regression model based on blobness, internal echo, and morphology features were implemented. A tumor candidate was then considered tumor when it reached a specific threshold of 0.4 for tumor likelihood. The proposed system showed a sensitivity rate of 100%, 90%, and 70% with false positives per pass of 17.4, 8.8, and 2.7.

**Table 1.1** Analysis of Image Processing Operations Applied for Bacterial Colony Detection and Counting

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Ref** | **Dataset** | **Colonies** | **Preprocessing** | **Segmentation** | **Edge Detection** | **Feature Extraction** | **Color Space** |
| [6] | 25 images | E. Coli DH5a | Rolling Ball Algorithm, ImageJ Brightness/Contrast Command, Median Filtering | ImageJ 'Find Maxima' Command | None | Colony Count | RGB, Grayscale |
| [9] | 3 samples | E.coli, P. aeruginosa,  E. faecalis | Noise Removal (Not Specified) | Binary Thresholding, Circular Hough Transform | Not Specified | Colony Count | RGB, Grayscale |
| [7] | Not Specified | Bacteria in soil from Mols Bjerge National Park | Illumination Correction (Not Specified) | Binary Thresholding, k-Means Clustering | None | Centroid, Area, Diameter, Perimeter, Circularity, Eccentricity, Convex Hull | RGB, Grayscale |
| [10] | 100 images | Not Specified | Noise Removal with Erosion | SIS Thresholding, DBSCAN, Circular Hough Transform | Not Specified | Colony Count | RGB, Grayscale |
| [8] | 108 images | Not Specified | None | Unsupervised U-Net Convolutional Neural Network | None | Colony Count | RGB |
| [13] | 15 images | C. albicans | None | Binary Thresholding, Watershed | None | Colony Count | RGB, Grayscale |
|  | 400 | Pathological Images, cell images, fluorescence imcroscopy | Image Smoothing Algorithm via Laplacian of Gaussian | Hessian analysis | None | Hessian-based Laplacian of Gaussian | 2D Grayscale Images |
|  | 173 | 136 breast lesion (58 benign lesions and 78 malignant lesions) and 37 normal cases | speckle noise reduction | None | None | Hessian analysis with multi-scale blob detection | Not Specified |
|  | 108 ALL-IDB,  367 SMC-IDB,  195 IUMS-IDB | White Blood Cells | Hue Thresholding,  Boolean Hue Mask | Watershed and Otsu | None | Convolutional Neural Network | RGB, HVS |

With this, the algorithms analyzed above are those that can be significant as a basis for the proposed framework. Articles found to have good colliding colony detection [6, 9, 10] were found to have difficulty in detecting translucent colonies. In contrast, the article found to have good translucent colony detection [7] uses a specific lighting and position setup. Likewise, most of the rest of the articles also use some specialized equipment and specific positioning in producing the image to be fed into their respective methodologies [6, 8, 9, 10, 13].

Consolidating some of the techniques found in these articles could solve issues coming from whichever category. Articles that make use of traditional image processing algorithms [6, 7, 9, 10] start out with some enhancement during preprocessing. Since most of these suffer reduced performance in detecting translucent colonies, histogram equalization will be used to improve the contrast to better highlight these colonies. Then, segmentation can be done though an iterative thresholding algorithm similar to that used by Marquard [10], which has the advantage of automatically determining a threshold value for grouping together a set of pixels within a range of intensity values. Noise reduction is also a recurring step [6, 9, 10] which can be done through different types of kernels. For this, erosion and dilation are chosen as it has an advantage of being able to separate the colonies from each other at the cost of morphological information which is not of concern. Finally, the algorithms which will be detecting and counting the colonies will be taken from the studies done by Kis [9], Marquard [10], and Grossi [29], which is the Circular Hough Transform and Blob Detection algorithm.

**Table 1.2** Algorithm Thresholding

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Ref** | **Algorithms** | **Threshold** |
| 1 | [26] | Histogram Equalization | Gray Intensity from 0 to 255 |
| 2 | [10] | Morphological Operations | Window Size of 3 |
| 3 | [27] | Non-Local Means Denoising | Filter Strength (h) of 31 |
| 4 | [10] | Circular Hough Transform (for petri dish bounds) | Radius from 1/6 to 1/2 of the image’s height |
| 5 | [29]  [30] | Blob Detection | Minimum Area of 5% of the image’s height  Minimum Inertia Ratio of 0.5  Minimum Convexity of 0.9 |

**Chapter 3**

# METHODOLOGY

## Methodology

In this section of the paper, the researchers will explain the detailed process and implementation of different algorithms in aiding bacterial colony counting which can improve the performance of detecting vague colonies as well as colliding colonies. This methodology was based initially on the methodology of Kiş et al. [9] with the addition of preprocessing techniques and segmentation to aid and enhance the performance of Circular Hough Transform in bacterial colony counting. However, after much difficulty in setting the parameters of Circular Hough Transform, this was replaced by Blob Detection. Thus, three (3) sets of algorithms will be analyzed and compared, specifically:

1. Circular Hough Transform with the applied preprocessing done by Kiş et al. [9]
2. Circular Hough Transform with an alternative set of preprocessing operations
3. Blob Detection with an alternative set of preprocessing operations.

The specification of the machine used in this study is shown in Table 1.1. The researchers will use Python for the programming language and will use Jupyter Notebook as editor in applying the sequence of algorithms needed for the study.

**Table 2.1** Machine Specification

|  |  |
| --- | --- |
| **Processor** | AMD Ryzen 7 3700X 8-Core Processor 3.60 GHz |
| **RAM** | 16GB |
| **System type** | 64-bit operating system |
| **Operating System** | Windows 10 |
| **Graphics Card** | GeForce RTX 3060 Ti |

## Conceptual Framework

The sample images that will be used in the research are acquired from the AGAR dataset, specifically, the researchers aim to only use samples that contain translucent colonies. The images will be converted to grayscale to avoid issues regarding various colors. This is done using the blue channel because the shadows that the colonies create are much easier to see in this channel. Afterwards, the petri dish bounds are detected using Circular Hough Transform. From here, the method employed by Kiş et. al [9] is used—the grayscale image is converted into binary, and the image is denoised via morphological transformations. Canny Edge Detection is then performed to extract the edges of each candidate colony. After preparing the sample images, the Circular Hough Transform algorithm will be used again for the feature extraction and the count will be determined. The alternative methods branch out after the bounds are detected, wherein Histogram Equalization will then be done within the bounds to adjust the intensity of the image and enhance its contrast. This is then followed by Non-Local Means Denoising to remove salt and pepper noise caused by the equalization process. Finally, both Circular Hough Transform and Blob Detection is used to detect and count the colonies in the image. Once the extraction is finished, testing and comparison will be performed next in acquiring data. The summary of the framework is shown in Figure 1.1.

Diagram

Description automatically generated

**Figure 1.1** Conceptual Framework

## Data Acquisition

The dataset that the researchers will use for the study are from the AGAR consist of 18,000 petri dishes pictures variating from having a homogenous or heterogenous type of bacterial colony dataset which is a publicly available dataset that only requires the user to provide an email for the download link of the dataset to be sent towards.

Diagram

Description automatically generated

**Figure 1.2** Data Acquisition Process Flow

## Dataset

The images of the petri dishes have 4 categorical elements. First is the background of the petri dish, in the selected dataset, the petri dish is set in a black background which contrast the color of the agar. The second is the foreground which is the agar where the bacterial colony are cultivated, the agar used for the cultivation of the bacteria have a yellowish-brown appearance. Next is the text in the petri dish indicating the company, expiration date, and the petri dish lot number. And lastly is the colony with the translucent appearance shown in Figure 1.3

A white plate with writing on it

Description automatically generated with medium confidenceA picture containing whiteboard

Description automatically generated

**Figure 1.3** Sample Containing Translucent Colonies (Marked Manually)

Source: AGAR Dataset

## Dataset Annotated Values

The sample images of the data set contain three annotations for the image, the background type or the color of the bacterial colony, the class of the bacterial colonies within the petri dish, and the counted colonies. The background has four categories: dark, bright, and vague, for this study, the group will only be focusing on identifying the colony number for the “vague” category for the background. The class contains the classification of the bacterial colony that is cultured in the petri dish. Lasty is the counted colonies which annotates the number of counted colonies and the class, height, width, id, and coordinate.

A picture containing text, indoor

Description automatically generatedA picture containing white, dishware

Description automatically generatedA white plate with writing on it

Description automatically generated with low confidence

**Figure 1.4** Samples Categorized as Dark (Left), Bright (Middle), and Vague (Right)

## Dataset Thresholding

From the said dataset, the researchers plan to explore only the homogenous bacterial colonies that have a translucent or opaque nature which can be challenging to detect even in the naked eye. The dataset will be filtered using the provided annotated values which tells the researcher if the colony is under the category of opaque bacterial colonies.

**Grayscale Conversion and Petri Dish Bounds Detection**

A picture containing text, device

Description automatically generatedThe images will be first converted into grayscale using the blue channel for simplicity and to further make translucent colonies easier to detect. The bounds of the petri dish is then detected to aid in telling whether the algorithms did in fact detect a colony that is within the dish and not outside of its area. This is done using the Circular Hough Transform, of which its parameters can be determined based on a percentage of the size of the image provided. Specifically, the algorithm will be configured to search for circles that have a radius of at least a sixth and at most half the image’s width. The largest detected circle is the petri dish bounds.

**Figure 1.5** Grayscale Image and Petri Dish Bounds Detection

## Initial Procedures on Detecting Colonies

**Binary Conversion**

Following the method by Kis et al. [9] the image is processed through a thresholding method which creates only two classes as opposed to multiple. The thresholding algorithm to be employed is Otsu’s method. It functions by iterating through every possible threshold value from 0 to 255 and selects the value that results in the highest between-class variance as the threshold. The between-class variance is defined as follows:

(1.1)

In this formula, σ2B is the between-class variance of a certain threshold value; Wb and Wf are the weights or percentage of pixels that are in the background and foreground; and b and f are the mean intensity values of the background and foreground, respectively. For this scenario, because there are pixels outside of the petri dish, the pixels to be considered in this algorithm must only be those that are within the bounds detected in the previous step.

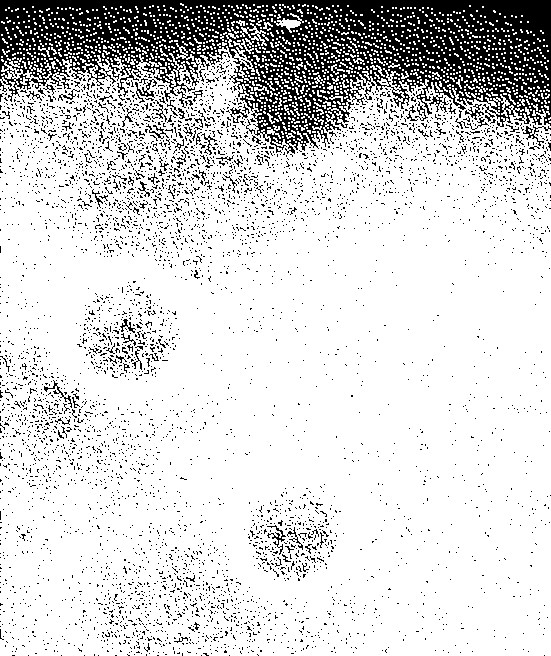
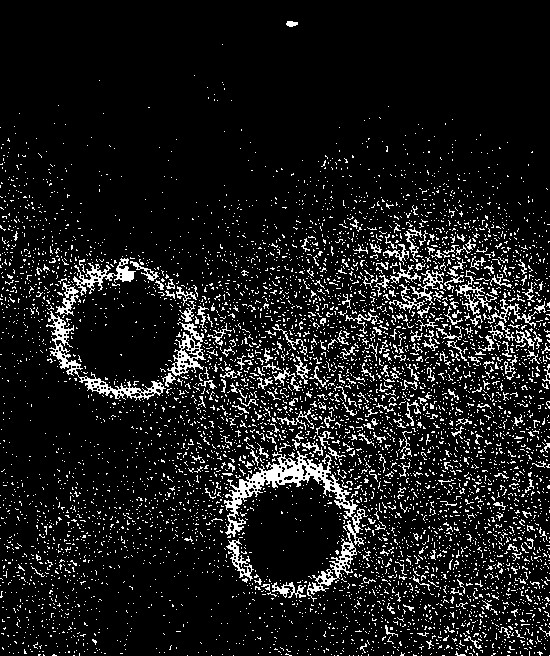
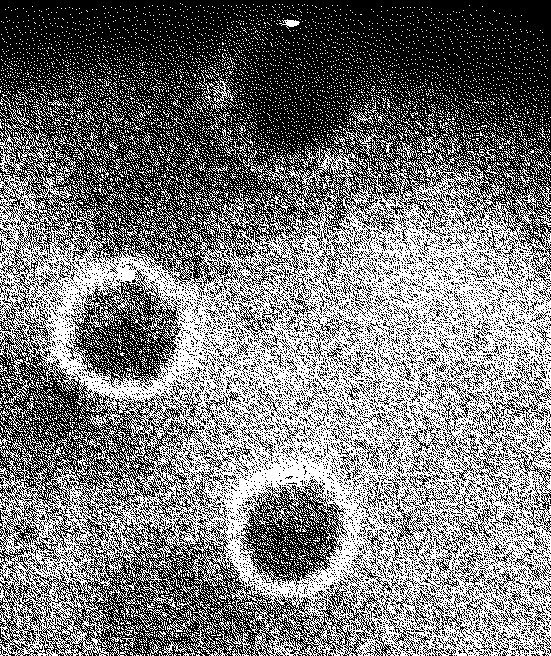
**Morphological Processing**

Erosion and dilation morphological processing techniques are then applied to remove some noise from the image, which can aid the Circular Hough Transform later Both erosion and dilation make use of a kernel which is a matrix that slides through the image.

A picture containing calendar

Description automatically generated (1.2)

For example, an erosion that makes use of a 3x3 kernel, while sliding through an image, will change its anchor pixel intensity into that of the lowest intensity that is found within its scope. In contrast, dilation changes the anchor pixel intensity into the highest intensity that is found. As a result, darker areas grow using erosion and brighter areas grow using dilation.



**Figure 1.6** Binary Image (left), Erosion (center), Dilation (right)

Previous studies have been shown to use little erosion and dilation [10] or none at all [6, 7, 8, 9], sometimes to retain morphological information [7, 10]. Since the colony count is the only feature of interest in this paper, higher values can be used so long as the entire colony does not disappear.

### **Colony Edge Filtering**

In preparation for the Circular Hough Transform, the Canny Edge Detector is used to filter the edge pixels found in the image, as these will be the pixels used by the algorithm later in counting. It functions first by sliding two kernels containing the Sobel operator, one oriented in the x-direction Gx and another oriented in the y-direction Gy.

A picture containing text, clock, receipt

Description automatically generated (1.3)

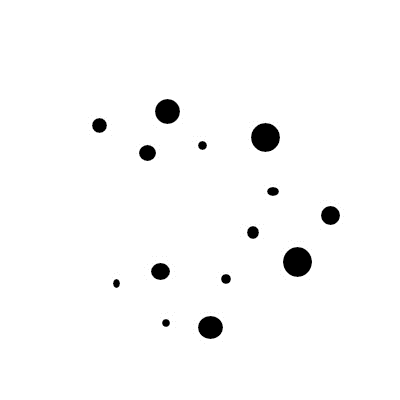
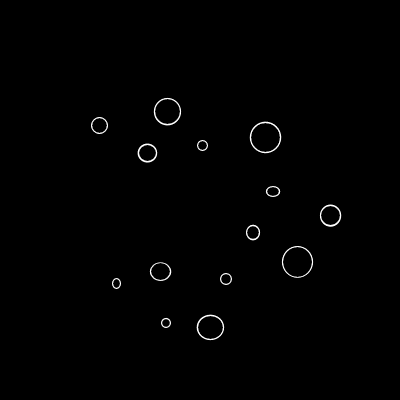
In each movement of the kernels, gray-level intensities which coincide with the kernel are multiplied to the constants and its summation yields the gradient approximation for that kernel orientation, this operation is referred to as convolution. The exact gradient magnitude is the vector sum of the approximations of both kernels, which replaces the value of the pixel at the center of the kernel. The direction of the gradient is also determined by getting the inverse tangent of the ratio between Gx and Gy.

(1.4)

(1.5)

Afterwards, non-maximum suppression is performed. This refers to determining if a pixel is a local maxima and removing or “suppressing” its neighboring pixels. The neighboring pixels are those pixels that are normal to the direction of the gradient of the target pixel. This causes the edges to become only 1-pixel wide.

Finally, a method called hysteresis thresholding is applied to determine which edges are useful or not. Two thresholds are applied in this process, the higher of which immediately determines that all pixels above its intensity level are useful. The lower threshold determines that the pixels between it and the higher threshold are useful only if they are connected to a pixel that is already above the higher threshold, the results of which are a clean binary image consisting of edges as white pixels and the background as black pixels.



**Figure 1.7** Visualization of the Sobel Operator Convolution and Hysteresis Thresholding

### **Circle Detection**

The Circular Hough Transform is the main algorithm that will be used in detecting and counting the number of colonies in the image. It applies the standard equation of a circle in determining the center and size of different circles within the image.

(1.6)

For each pixel (x, y) of a candidate colony, this algorithm draws circles of increasing radii on a three-dimensional plane above the image called an accumulator, forming a cone shape with each level corresponding to a radius length which starts from 1 up to a specified or formulated maximum based on the size of the image itself. Afterwards, for each radius length, the pixels that intersect with the edges of the drawn circles are incremented in its respective accumulator matrix. The coordinate with the highest value among each level of the accumulator is determined to be the center (a, b) and its radius is the level that it is found in, which are the features to be used in counting.

For visualization purposes, Figure 1.8 illustrates the Circular Hough Transform with three edge pixels highlighted in red wherein using r1 does not detect the circle due to the tie in the number of intersections highlighted in yellow and using r2 does detect the circle due to the pixel highlighted in green having the most intersections.

A picture containing sport, athletic game

Description automatically generated

**Figure 1.8** Circular Hough Transform using Three Edge Pixels

A picture containing dishware, tableware, pan

Description automatically generated

**Figure 1.9** Results of the Circular Hough Transform drawn on a Sample Image

Source: Kis et al. (2019) [9]

## Alternative Procedures on Detecting Colonies

### **Contrast Improvement**

For the alternative method, histogram equalization, which was used by Byun et al. [27] is performed on the grayscale image to improve its contrast. This refers to remapping the intensities of each pixel using the cumulative frequency distribution of the image’s gray-level intensities.

Logo, icon, company name

Description automatically generated

**Figure 1.10** Example of a Cumulative Frequency Distribution

Source: OpenCV Development Team (2021)

For example, an image can have the cumulative frequency distribution shown in Figure 1.11, which has a range of 0 to 255 which represents valid intensity values, and a domain of 0 to 1 which represents the percentage. A pixel with an intensity level ‘x’ is remapped to have a new intensity level which is equal to the maximum intensity value multiplied by the factor ‘y’. This results in a histogram whose intensity levels are spread out throughout its range of values. The formal definition of histogram equalization is as follows:

(1.7)

Histogram

Description automatically generated with medium confidence

**Figure 1.11** Effect of Histogram Equalization on Frequency Distribution

Source: OpenCV Development Team (2021)

A close-up of the moon

Description automatically generated with medium confidence

**Figure 1.12** Histogram Equalization Visualization

This is often a great method of enhancing the contrast in an image, which eliminate the need for predefined or “black box” setups [7, 8, 10, 15]. However, images provided by the AGAR dataset have a black background which occupies a large percentage of the gray-level intensities in the image. Performing histogram equalization on this type of image will cause only the contrast in the background to be enhanced. Therefore, the pixels to be considered in the cumulative frequency distribution must only be the pixels that are within the petri dish bounds, which should be detected by the previous step, similar to that used for the binary thresholding algorithm.

### **Denoising**

Since Histogram Equalization often creates a large amount of noise, it is necessary to denoise the image afterwards in order to prepare it for feature extraction. For this, Non-Local Means Denoising, which was used by Yang et al. [28] is used due to its ability to make objects take on a solid shade without being blurred to preserve edges. This algorithm smooths objects by using the weighted mean intensity, based on the similarity using the mean squared error, of all patches in the image that are the same size as the kernel that is being denoised.

A close-up of the moon

Description automatically generated with medium confidence

**Figure 1.13** Non-Local Means Denoising

### **Blob Detection**

Blob Detection was then considered as an alternative method for counting the colonies, which was determined to be more appropriate due to the fact that it was not dependent on edge detection and instead worked by grouping together pixels of the same color range in an iterative binarization process given the minimum and maximum threshold range. These are then filtered based on different parameters such as area, circularity, inertia, and convexity. These parameters are much more versatile compared to that of the Circular Hough Transform because the prerequisite edge detection may not give reliable results when the gradient between the colonies and the background is gradual. It was also found that parameters such as the minimum area and maximum area of blobs could be given large threshold ranges and still give relatively more precise results when glanced upon.

A picture containing text, indoor, different

Description automatically generated

**Figure 1.14** Blob Detection Results

## Colony Counting

 Each circle found by the Circular Hough Transform and Blob Detection algorithms is labeled as a bacteria colony. For each circle found, an accumulator is incremented which determines the number of colonies in the image.

## Testing

Since there are parameters that must be set prior to processing the images, a given range for each of these must be set for each method. Each combination of the values within these ranges must be tested and its performance metrics must be compared. The average precision, recall, and F-score of all the images for a given set of parameters will be computed, and the results of the set that gives the highest average F-score will be chosen as the point of comparison for that method against the other methods.

Also, an Independent Sample T-test will be implemented to measure if there is a significant difference between the counted colonies using either Circular Hough Transform or Blob Detection and the actual count of the colony. First the researchers will create hypothesis that will be proven by the t-test, the first is the null hypothesis where it states that there is no significant difference between the automated count and the actual count (µ1 = µ2), then the second hypothesis would be the alternative hypothesis which states that there is a significant difference between the two counts (µ1 ≠ µ2).

After stating the two hypotheses, the researchers will then get the mean average of the two datasets using the formula in Formula 1.9 where ΣX is the summation of all the values in the data

and N is the size of the data.

Text

Description automatically generated (1.8)

After getting the mean the researchers will then solve for the variance of the two datasets using the formula stated in Formula 1.10.

A picture containing table

Description automatically generated (1.9)

Given the variance, the researchers will then be able to compute for Standard Error of

Difference Between Means of the two datasets using the formula indicated in Equation 1.10.

Text, letter

Description automatically generated (1.10)

After computing for the Standard Error of the Difference Between Means, the t-statistics can then be computed. T-statistics is the quotient of the difference between the mean of the first and second dataset, and the standard error of the difference between means.

Text

Description automatically generated (1.11)

To conclude if the difference is significant or not, the researchers will first assign an α of

0.5 and compute the degree of freedom using the formula indicated in Equation 1.12.

Text, letter

Description automatically generated (1.12)

Using the t-table, the researchers will then find the critical value based on the given alpha and the computed df.

Table

Description automatically generated

**Figure 1.15** T-table

Source: Lecky, F. (2001) [25]

The critical value from the t-table will be compared to the computed t-statistics. If the t-statistics is greater than the critical value, the null hypothesis would be rejected, meaning that there is a significant difference between the automated count and the actual count. However, if the t-statistics is less than the critical value, the null hypothesis would be accepted, concluding that there is no significant difference in the two datasets, implying that the automated count is as good as manually counting can be.

**CHAPTER 4**

# RESULTS AND DISCUSSION

## Kis Implementation of Circular Hough Transform

Prior to feeding the images into the main algorithm, the images were first converted into greyscale using blue channel to make translucent colonies more identifiable as seen in Figure 4.1. After which, the bounds of the petri dish were detected using Circular Hough Transform to ensure that the main algorithm will only operates inside the petri dish, this can be seen in Figure 4.2



**Figure 4.1** Grayscale Image **Figure 4.2** Petri Dish Bounds Detection

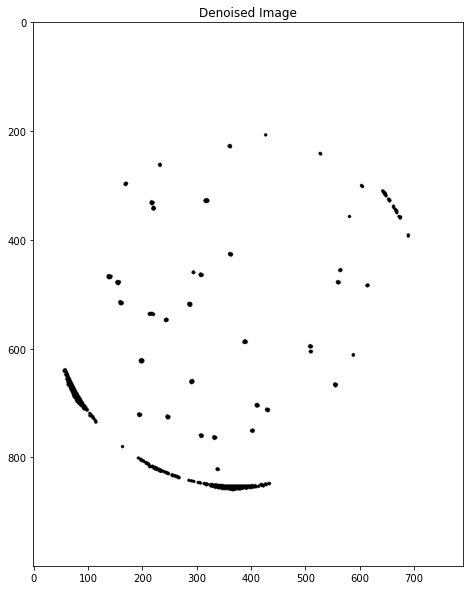
In Figure 4.3, the types of bacterial colony that is present in the petri dish can be seen clearly. The red circles are the translucent colonies, and the green circles are the opaque colonies. It is also noticeable that the size of the two colonies differs, having translucent colonies larger than the opaque colonies. This difference would be used in the succeeding algorithms to limit detection errors that are caused by setting a large gap between the minimum and maximum size of the colonies to be detected.

A picture containing diagram

Description automatically generated

**Figure 4.3** Translucent (Red) and Opaque (Green) Bacterial Colony

Following the process implemented by Kis et al [9], the grayscale image in then binarized, this can be seen in Figure 4.4 where the image is simplified in two colors namely black and white. After the binarization of the image, it was then denoised via morphological operations to eliminate noise and further characterized the colonies in the petri dish. In Figure 4.5, it is noticeable that the image is much cleaner and do not have the text and the boundary of the petri dish.

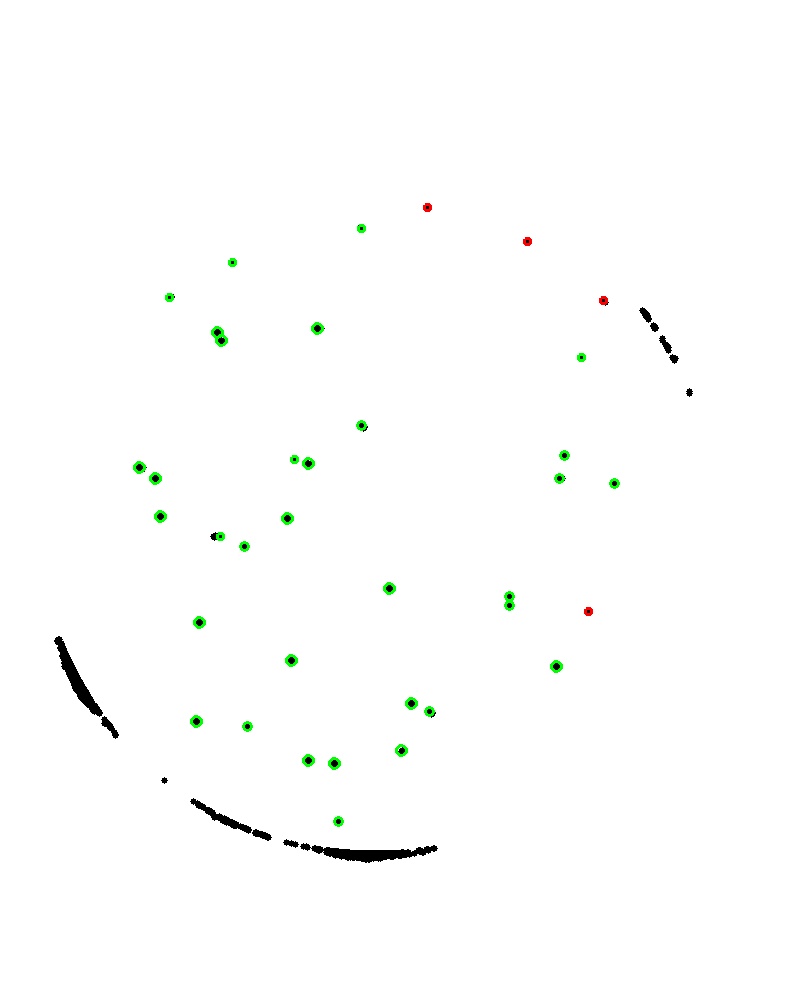


**Figure 4.4** Binarized Image **Figure 4.5** Denoised Image

The resulting images are then fed through the Circular Hough Transform using a range of parameters shown in Table 4.1 which are only for detecting opaque colonies since the binarization method for the study being replicated causes the translucent colonies to be removed as shown by comparing Figure 4.3 and Figure 4.4. The set of parameter values which gives the highest average F-score for all images was then used as the final parameters to be used for comparisons.

**Table 4.1** Range of Parameters for the Circular Hough Transform using the Study of Kis et al. [9]

|  |  |  |
| --- | --- | --- |
|  | **Opaque Colonies** | |
|  | **Range (Step)** | **Final Value** |
| Minimum Distance Between Centers | 1-6 (2) pixels | 5 pixels |
| Accumulator | 1-8 (1) intersections | 6 intersections |
| Minimum Radius | 1-6 (2) pixels | 1 pixel |
| Maximum Radius | 1-8 (2) pixels | 5 pixels |



**Figure 4.6** Colony Detection on a Binary Image using the Circular Hough Transform

**Alternative Process**

For the alternative process, the preprocessing techniques used by Kis et al [9] in their implementation of Circular Hough Transform were changed to increase detection capabilities for translucent colonies. This preprocessing will also be used in the implementation of the Blob Detection algorithm. Like the first implementation, the image is first converted into grayscale using the blue channel but is then followed by histogram equalization to improve the contrast of the image which can be seen in Figure 4.7. Afterwards, Non-Local Means Denoising is applied to remove noise in the image. Applying this step in the preprocessing preserves the edges and the color of the object in the image, making the colonies more consistently close to each other in terms of intensity which can be seen in Figure 4.8.



**Figure 4.7** Histogram Equalization **Figure 4.8** Non-Local Means Denoising

The proposed preprocessing techniques to increase the detection capability of translucent colonies will be both used in Circular Hough Transform and Blob Detection algorithm. Since the translucent colonies are preserved using this implementation, two sets of parameters for each algorithm will be used–one of which will be used for the opaque colonies, and the other instead for the translucent colonies. Like the initial process, a range of values will be tested and the set that yields the highest average F-score will be chosen for comparisons.

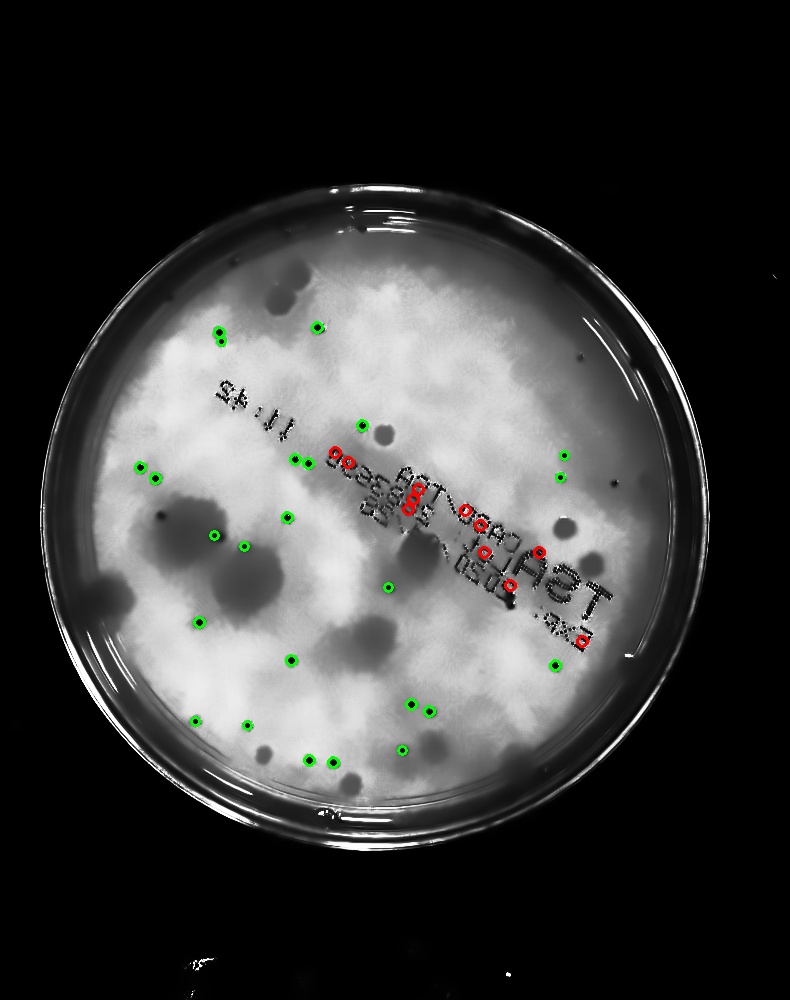
**Table 4.2** Range of Parameters for the Circular Hough Transform using the Alternative Process

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Opaque Colonies** | | **Translucent Colonies** | |
|  | **Range (Step)** | **Final Value** | **Range (Step)** | **Final Value** |
| Minimum Distance Between Centers | 1-6 (2) pixels | 5 pixels | 20-21 (1) pixels | 20 pixels |
| Accumulator | 1-8 (1) intersections | 7 intersections | 7-12 (1) intersections | 7 intersections |
| Minimum Radius | 1-6 (2) pixels | 3 pixels | 13-22 (2) pixels | 15 pixels |
| Maximum Radius | 1-8 (2) pixels | 3 pixels | 13-24 (2) pixels | 15 pixels |

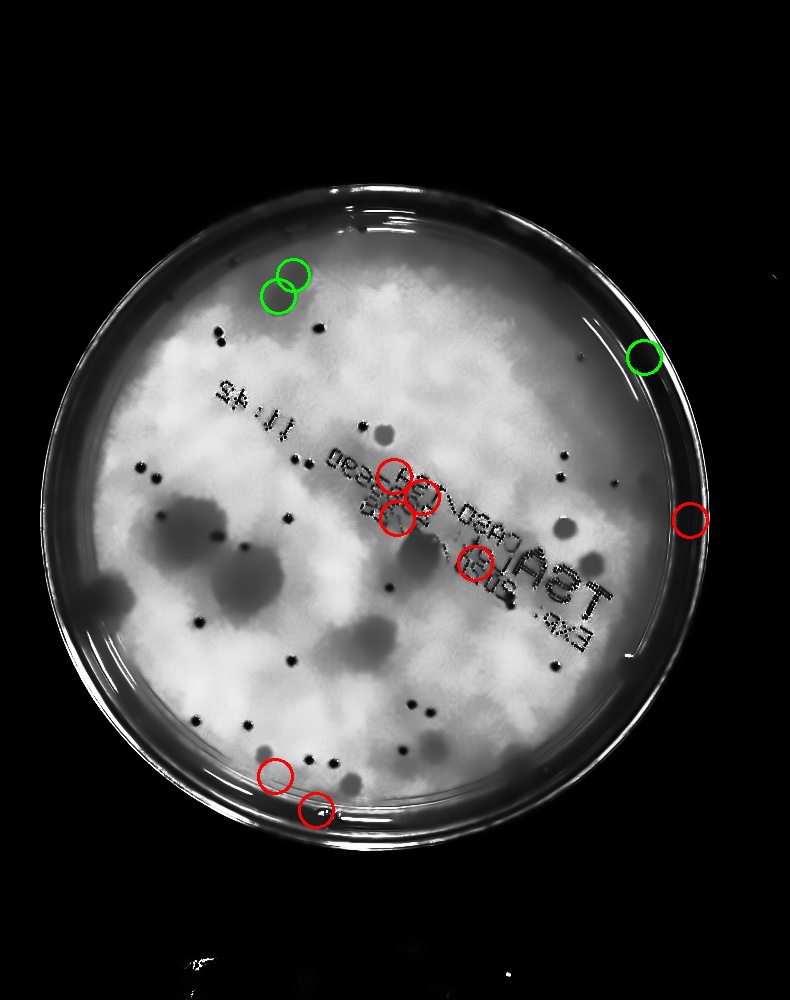
**Table 4.3** Range of Parameters for the Blob Detection Algorithm using the Alternative Process

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Opaque Colonies** | | **Translucent Colonies** | |
|  | **Range (Step)** | **Final Value** | **Range (Step)** | **Final Value** |
| Minimum Area |  | 45 pixels |  | 300 pixels |
| Maximum Area |  | 70 pixels |  |  |
| Minimum Repeatability | 2-4 (1) repetitions | 2 repetitions | 2-4 (1) repetitions | 2 repetitions |
| Minimum Distance Between Blobs | 2-3 (1) pixels | 3 pixels | 2-3 (1) pixels | 3 pixels |
| Minimum Inertia Ratio | 0.4-0.7 (0.1) | 0.4 | 0.4-0.7 (0.1) | 0.4 |
| Minimum Convexity | 0.7-0.9 (0.1) | 0.9 | 0.7-0.9 (0.1) | 0.9 |

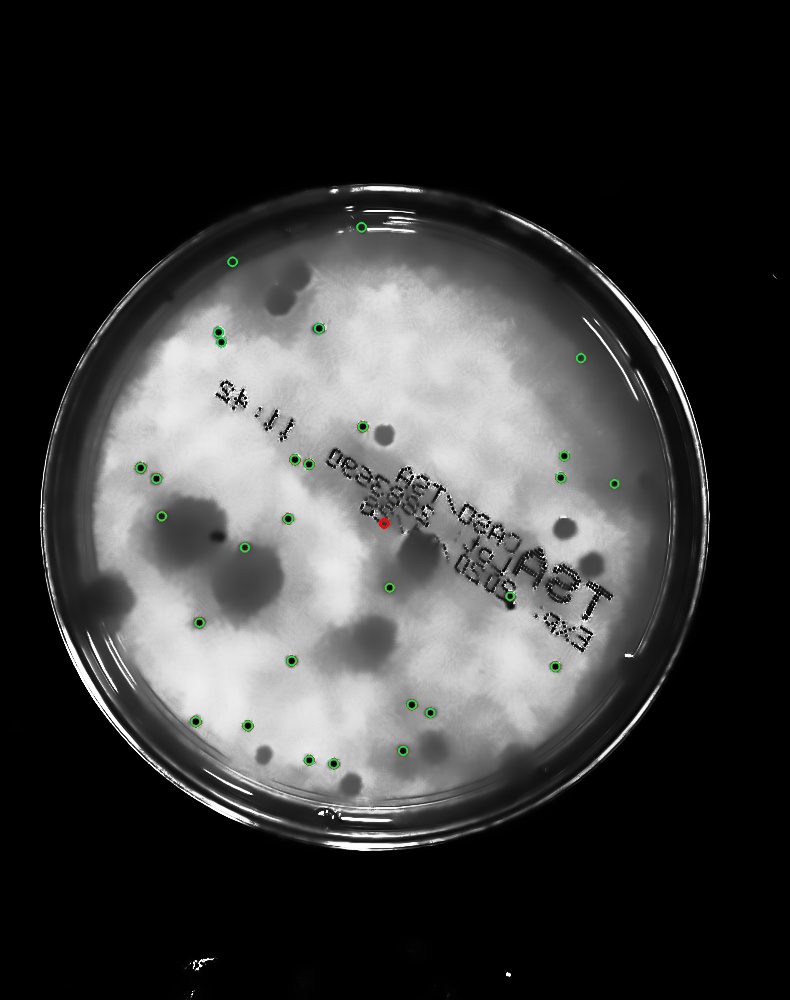
One of the resulting images of the Circular Hough Transform can be seen in Figure 4.9 and Figure 4.10 while that of Blob Detection is seen in Figure 4.11 and Figure 4.12.



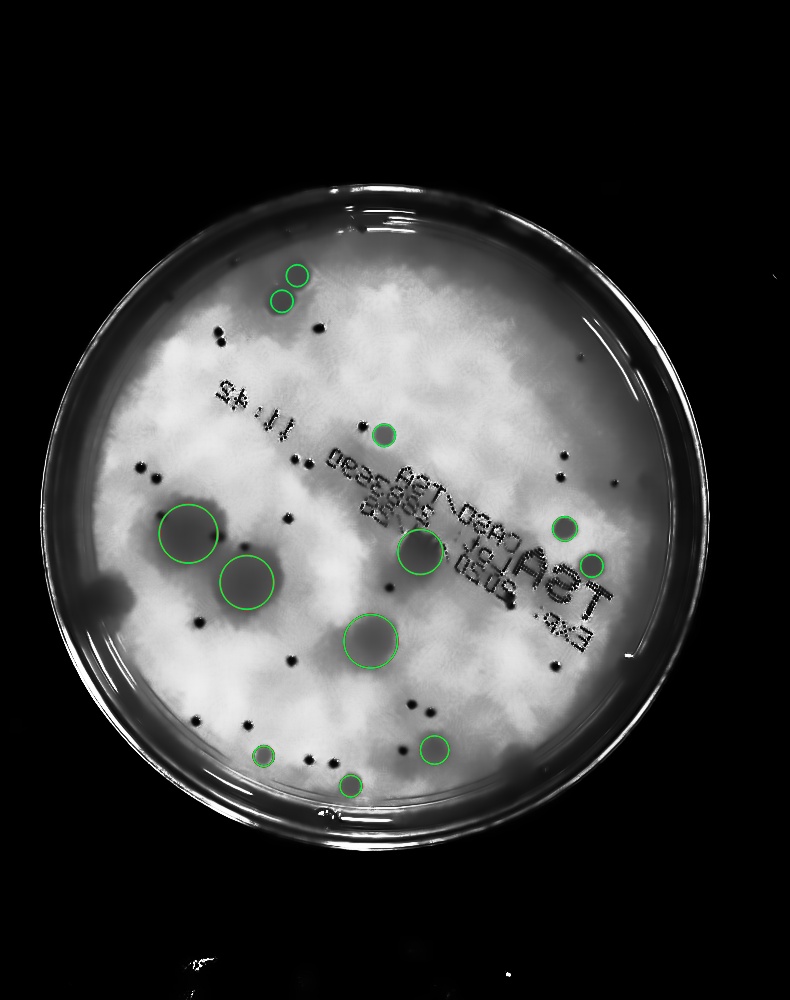
**Figure 4.9** Opaque Colony Detection Using Circular Hough Transform

****

**Figure 4.10** Translucent Colony Detection Using Circular Hough Transform



**Figure 4.11** Opaque Colony Detection Using Blob Detection Algorithm



**Figure 4.12** Translucent Colony Detection Using Blob Detection Algorithm

**Table 4.5** Opaque Colony Count Results for the Process of Kis et al. [9]

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID | F-Score | Precision | Recall | Actual | Counted | TP | FP | FN |
| 12442.jpg | **91.67%** | 89.19% | 94.29% | 39 | 37 | 33 | 4 | 2 |
| 12444.jpg | **43.90%** | 90.00% | 29.03% | 32 | 10 | 9 | 1 | 22 |
| 12452.jpg | **78.57%** | 91.67% | 68.75% | 17 | 12 | 11 | 1 | 5 |
| 12454.jpg | **84.21%** | 72.73% | 100.00% | 10 | 11 | 8 | 3 | 0 |
| 12455.jpg | **90.48%** | 86.36% | 95.00% | 23 | 22 | 19 | 3 | 1 |
| 12456.jpg | **40.00%** | 25.00% | 100.00% | 6 | 20 | 5 | 15 | 0 |
| 12457.jpg | **55.00%** | 100.00% | 37.93% | 29 | 11 | 11 | 0 | 18 |
| 12460.jpg | **73.68%** | 63.64% | 87.50% | 12 | 11 | 7 | 4 | 1 |
| 12461.jpg | **66.67%** | 71.43% | 62.50% | 10 | 7 | 5 | 2 | 3 |
| 12463.jpg | **60.00%** | 42.86% | 100.00% | 21 | 28 | 12 | 16 | 0 |
| 12465.jpg | **50.00%** | 45.83% | 55.00% | 33 | 24 | 11 | 13 | 9 |
| 12466.jpg | **77.27%** | 77.27% | 77.27% | 27 | 22 | 17 | 5 | 5 |
| 12470.jpg | **62.50%** | 100.00% | 45.45% | 22 | 10 | 10 | 0 | 12 |
| 12471.jpg | **0.00%** | 0.00% | 0.00% | 29 | 2 | 0 | 2 | 27 |
| 12475.jpg | **57.47%** | 60.98% | 54.35% | 62 | 41 | 25 | 16 | 21 |
| 12476.jpg | **79.17%** | 65.52% | 100.00% | 24 | 29 | 19 | 10 | 0 |
| 12478.jpg | **69.23%** | 75.00% | 64.29% | 17 | 12 | 9 | 3 | 5 |
| 12479.jpg | **97.73%** | 97.73% | 97.73% | 45 | 44 | 43 | 1 | 1 |
| 12480.jpg | **33.33%** | 80.00% | 21.05% | 20 | 5 | 4 | 1 | 15 |
| 12481.jpg | **61.54%** | 100.00% | 44.44% | 27 | 12 | 12 | 0 | 15 |
| 12483.jpg | **48.15%** | 70.27% | 36.62% | 82 | 37 | 26 | 11 | 45 |
| 12489.jpg | **50.00%** | 33.33% | 100.00% | 18 | 21 | 7 | 14 | 0 |
| 12490.jpg | **85.71%** | 75.00% | 100.00% | 25 | 28 | 21 | 7 | 0 |
| 12492.jpg | **44.44%** | 28.57% | 100.00% | 10 | 14 | 4 | 10 | 0 |
| 12495.jpg | **89.80%** | 81.48% | 100.00% | 26 | 27 | 22 | 5 | 0 |
| 12497.jpg | **76.47%** | 76.47% | 76.47% | 42 | 34 | 26 | 8 | 8 |
| 12500.jpg | **21.43%** | 100.00% | 12.00% | 50 | 6 | 6 | 0 | 44 |
| 12503.jpg | **91.30%** | 87.50% | 95.45% | 25 | 24 | 21 | 3 | 1 |
| 12505.jpg | **75.56%** | 60.71% | 100.00% | 26 | 28 | 17 | 11 | 0 |
| 12507.jpg | **36.36%** | 100.00% | 22.22% | 18 | 4 | 4 | 0 | 14 |

**Table 4.6** Opaque Colony Counts for the Alternative Process using Circular Hough Transform

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID | F-Score | Precision | Recall | Actual | Counted | TP | FP | FN |
| 12442.jpg | **76.19%** | 68.57% | 85.71% | 39 | 35 | 24 | 11 | 4 |
| 12444.jpg | **17.14%** | 13.04% | 25.00% | 32 | 23 | 3 | 20 | 9 |
| 12452.jpg | **66.67%** | 50.00% | 100.00% | 17 | 18 | 9 | 9 | 0 |
| 12454.jpg | **58.82%** | 41.67% | 100.00% | 10 | 12 | 5 | 7 | 0 |
| 12455.jpg | **64.86%** | 48.00% | 100.00% | 23 | 25 | 12 | 13 | 0 |
| 12456.jpg | **29.63%** | 17.39% | 100.00% | 6 | 23 | 4 | 19 | 0 |
| 12457.jpg | **73.91%** | 65.38% | 85.00% | 29 | 26 | 17 | 9 | 3 |
| 12460.jpg | **55.17%** | 38.10% | 100.00% | 12 | 21 | 8 | 13 | 0 |
| 12461.jpg | **62.50%** | 45.45% | 100.00% | 10 | 11 | 5 | 6 | 0 |
| 12463.jpg | **68.57%** | 52.17% | 100.00% | 21 | 23 | 12 | 11 | 0 |
| 12465.jpg | **39.02%** | 38.10% | 40.00% | 33 | 21 | 8 | 13 | 12 |
| 12466.jpg | **71.43%** | 55.56% | 100.00% | 27 | 27 | 15 | 12 | 0 |
| 12470.jpg | **62.50%** | 47.62% | 90.91% | 22 | 21 | 10 | 11 | 1 |
| 12471.jpg | **6.67%** | 6.25% | 7.14% | 29 | 16 | 1 | 15 | 13 |
| 12475.jpg | **54.12%** | 71.88% | 43.40% | 62 | 32 | 23 | 9 | 30 |
| 12476.jpg | **80.00%** | 66.67% | 100.00% | 24 | 27 | 18 | 9 | 0 |
| 12478.jpg | **53.33%** | 36.36% | 100.00% | 17 | 22 | 8 | 14 | 0 |
| 12479.jpg | **75.00%** | 79.41% | 71.05% | 45 | 34 | 27 | 7 | 11 |
| 12480.jpg | **33.33%** | 28.57% | 40.00% | 20 | 14 | 4 | 10 | 6 |
| 12481.jpg | **77.27%** | 68.00% | 89.47% | 27 | 25 | 17 | 8 | 2 |
| 12483.jpg | **52.25%** | 74.36% | 40.28% | 82 | 39 | 29 | 10 | 43 |
| 12489.jpg | **43.48%** | 33.33% | 62.50% | 18 | 15 | 5 | 10 | 3 |
| 12490.jpg | **64.86%** | 52.17% | 85.71% | 25 | 23 | 12 | 11 | 2 |
| 12492.jpg | **33.33%** | 20.00% | 100.00% | 10 | 10 | 2 | 8 | 0 |
| 12495.jpg | **85.11%** | 74.07% | 100.00% | 26 | 27 | 20 | 7 | 0 |
| 12497.jpg | **72.73%** | 82.76% | 64.86% | 42 | 29 | 24 | 5 | 13 |
| 12500.jpg | **18.18%** | 20.83% | 16.13% | 50 | 24 | 5 | 19 | 26 |
| 12503.jpg | **76.00%** | 61.29% | 100.00% | 25 | 31 | 19 | 12 | 0 |
| 12505.jpg | **76.19%** | 66.67% | 88.89% | 26 | 24 | 16 | 8 | 2 |
| 12507.jpg | **50.00%** | 40.00% | 66.67% | 18 | 15 | 6 | 9 | 3 |

**Table 4.7** Opaque Colony Count for the Alternative Process using Blob Detection

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID | F-Score | Precision | Recall | Actual | Counted | TP | FP | FN |
| 12442.jpg | **85.29%** | 96.67% | 76.32% | 39 | 30 | 29 | 1 | 9 |
| 12444.jpg | **66.67%** | 94.12% | 51.61% | 32 | 17 | 16 | 1 | 15 |
| 12452.jpg | **82.76%** | 80.00% | 85.71% | 17 | 15 | 12 | 3 | 2 |
| 12454.jpg | **88.89%** | 100.00% | 80.00% | 10 | 8 | 8 | 0 | 2 |
| 12455.jpg | **93.02%** | 86.96% | 100.00% | 23 | 23 | 20 | 3 | 0 |
| 12456.jpg | **80.00%** | 100.00% | 66.67% | 6 | 4 | 4 | 0 | 2 |
| 12457.jpg | **84.00%** | 84.00% | 84.00% | 29 | 25 | 21 | 4 | 4 |
| 12460.jpg | **90.91%** | 83.33% | 100.00% | 12 | 12 | 10 | 2 | 0 |
| 12461.jpg | **82.35%** | 87.50% | 77.78% | 10 | 8 | 7 | 1 | 2 |
| 12463.jpg | **76.47%** | 92.86% | 65.00% | 21 | 14 | 13 | 1 | 7 |
| 12465.jpg | **82.14%** | 100.00% | 69.70% | 33 | 23 | 23 | 0 | 10 |
| 12466.jpg | **89.80%** | 100.00% | 81.48% | 27 | 22 | 22 | 0 | 5 |
| 12470.jpg | **84.21%** | 100.00% | 72.73% | 22 | 16 | 16 | 0 | 6 |
| 12471.jpg | **58.54%** | 92.31% | 42.86% | 29 | 13 | 12 | 1 | 16 |
| 12475.jpg | **69.47%** | 94.29% | 55.00% | 62 | 35 | 33 | 2 | 27 |
| 12476.jpg | **90.91%** | 95.24% | 86.96% | 24 | 21 | 20 | 1 | 3 |
| 12478.jpg | **82.76%** | 80.00% | 85.71% | 17 | 15 | 12 | 3 | 2 |
| 12479.jpg | **88.89%** | 100.00% | 80.00% | 45 | 36 | 36 | 0 | 9 |
| 12480.jpg | **75.00%** | 92.31% | 63.16% | 20 | 13 | 12 | 1 | 7 |
| 12481.jpg | **87.50%** | 100.00% | 77.78% | 27 | 21 | 21 | 0 | 6 |
| 12483.jpg | **72.87%** | 97.92% | 58.02% | 82 | 48 | 47 | 1 | 34 |
| 12489.jpg | **80.00%** | 100.00% | 66.67% | 18 | 12 | 12 | 0 | 6 |
| 12490.jpg | **80.95%** | 100.00% | 68.00% | 25 | 17 | 17 | 0 | 8 |
| 12492.jpg | **75.00%** | 100.00% | 60.00% | 10 | 6 | 6 | 0 | 4 |
| 12495.jpg | **89.36%** | 91.30% | 87.50% | 26 | 23 | 21 | 2 | 3 |
| 12497.jpg | **84.93%** | 93.94% | 77.50% | 42 | 33 | 31 | 2 | 9 |
| 12500.jpg | **64.86%** | 96.00% | 48.98% | 50 | 25 | 24 | 1 | 25 |
| 12503.jpg | **97.96%** | 100.00% | 96.00% | 25 | 24 | 24 | 0 | 1 |
| 12505.jpg | **81.82%** | 100.00% | 69.23% | 26 | 18 | 18 | 0 | 8 |
| 12507.jpg | **80.00%** | 100.00% | 66.67% | 18 | 12 | 12 | 0 | 6 |

**Table 4.8** Independent Sample T-Test Result for Opaque Colonies Only

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Actual | CHT (Kis) | CHT (Alternative) | BD (Alternative) |
| Mean | 27.56667 | 19.76667 | 23.1 | 19.63333 |
| Size | 30 | 30 | 30 | 30 |
| Variance | 984.7333 | 504.0667 | 560.4667 | 458.6 |
| df |  | 58 | 58 | 58 |
| Critical Value |  | 1.671 | 1.671 | 1.671 |
| SEDM |  | 7.165049 | 7.299504 | 7.054794 |
| t-statistic |  | **1.088618** | **0.611914** | **1.124531** |

For the independent sample t-test for the Opaque Colony count, the t-statistic of the three-counting process were less than the critical value of 1.671. Which means that there is no significant difference between the number of colonies that the counting process provided and the actual number of colonies in the image.

**Table 4.9** Translucent Colony Count Results for the Alternative Process using Circular Hough Transform

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID | F-Score | Precision | Recall | Actual | Counted | TP | FP | FN |
| 12442.jpg | **28.57%** | 30.00% | 27.27% | 18 | 10 | 3 | 7 | 8 |
| 12444.jpg | **58.82%** | 41.67% | 100.00% | 12 | 12 | 5 | 7 | 0 |
| 12452.jpg | **46.15%** | 33.33% | 75.00% | 20 | 18 | 6 | 12 | 2 |
| 12454.jpg | **0.00%** | 0.00% | 0.00% | 1 | 0 | 0 | 0 | 1 |
| 12455.jpg | **50.00%** | 47.06% | 53.33% | 24 | 17 | 8 | 9 | 7 |
| 12456.jpg | **0.00%** | 0.00% | 0.00% | 5 | 0 | 0 | 0 | 5 |
| 12457.jpg | **56.41%** | 55.00% | 57.89% | 28 | 20 | 11 | 9 | 8 |
| 12460.jpg | **50.00%** | 58.33% | 43.75% | 21 | 12 | 7 | 5 | 9 |
| 12461.jpg | **55.32%** | 48.15% | 65.00% | 34 | 27 | 13 | 14 | 7 |
| 12463.jpg | **21.05%** | 16.67% | 28.57% | 17 | 12 | 2 | 10 | 5 |
| 12465.jpg | **76.47%** | 61.90% | 100.00% | 17 | 21 | 13 | 8 | 0 |
| 12466.jpg | **14.29%** | 11.11% | 20.00% | 13 | 9 | 1 | 8 | 4 |
| 12470.jpg | **75.68%** | 70.00% | 82.35% | 23 | 20 | 14 | 6 | 3 |
| 12471.jpg | **81.08%** | 68.18% | 100.00% | 21 | 22 | 15 | 7 | 0 |
| 12475.jpg | **22.22%** | 22.22% | 22.22% | 16 | 9 | 2 | 7 | 7 |
| 12476.jpg | **26.67%** | 28.57% | 25.00% | 13 | 7 | 2 | 5 | 6 |
| 12478.jpg | **30.00%** | 25.00% | 37.50% | 17 | 12 | 3 | 9 | 5 |
| 12479.jpg | **35.29%** | 21.43% | 100.00% | 11 | 14 | 3 | 11 | 0 |
| 12480.jpg | **70.83%** | 73.91% | 68.00% | 31 | 23 | 17 | 6 | 8 |
| 12481.jpg | **45.45%** | 38.46% | 55.56% | 17 | 13 | 5 | 8 | 4 |
| 12483.jpg | **31.58%** | 25.00% | 42.86% | 16 | 12 | 3 | 9 | 4 |
| 12489.jpg | **56.00%** | 38.89% | 100.00% | 15 | 18 | 7 | 11 | 0 |
| 12490.jpg | **55.17%** | 44.44% | 72.73% | 21 | 18 | 8 | 10 | 3 |
| 12492.jpg | **80.00%** | 72.00% | 90.00% | 27 | 25 | 18 | 7 | 2 |
| 12495.jpg | **37.50%** | 23.08% | 100.00% | 11 | 13 | 3 | 10 | 0 |
| 12497.jpg | **50.00%** | 50.00% | 50.00% | 21 | 14 | 7 | 7 | 7 |
| 12500.jpg | **34.78%** | 28.57% | 44.44% | 19 | 14 | 4 | 10 | 5 |
| 12503.jpg | **62.50%** | 83.33% | 50.00% | 22 | 12 | 10 | 2 | 10 |
| 12505.jpg | **70.97%** | 57.89% | 91.67% | 20 | 19 | 11 | 8 | 1 |
| 12507.jpg | **75.00%** | 62.50% | 93.75% | 25 | 24 | 15 | 9 | 1 |

**Table 4.10** Translucent Colony Count for the Alternative Process using Blob Detection

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID | F-Score | Precision | Recall | Actual | Counted | TP | FP | FN |
| 12442.jpg | **80.00%** | 100.00% | 66.67% | 18 | 12 | 12 | 0 | 6 |
| 12444.jpg | **85.71%** | 81.82% | 90.00% | 12 | 11 | 9 | 2 | 1 |
| 12452.jpg | **66.67%** | 90.91% | 52.63% | 20 | 11 | 10 | 1 | 9 |
| 12454.jpg | **100.00%** | 100.00% | 100.00% | 1 | 1 | 1 | 0 | 0 |
| 12455.jpg | **70.27%** | 92.86% | 56.52% | 24 | 14 | 13 | 1 | 10 |
| 12456.jpg | **33.33%** | 50.00% | 25.00% | 5 | 2 | 1 | 1 | 3 |
| 12457.jpg | **63.41%** | 92.86% | 48.15% | 28 | 14 | 13 | 1 | 14 |
| 12460.jpg | **72.73%** | 92.31% | 60.00% | 21 | 13 | 12 | 1 | 8 |
| 12461.jpg | **69.23%** | 85.71% | 58.06% | 34 | 21 | 18 | 3 | 13 |
| 12463.jpg | **69.23%** | 90.00% | 56.25% | 17 | 10 | 9 | 1 | 7 |
| 12465.jpg | **78.57%** | 68.75% | 91.67% | 17 | 16 | 11 | 5 | 1 |
| 12466.jpg | **47.06%** | 66.67% | 36.36% | 13 | 6 | 4 | 2 | 7 |
| 12470.jpg | **68.57%** | 100.00% | 52.17% | 23 | 12 | 12 | 0 | 11 |
| 12471.jpg | **83.33%** | 100.00% | 71.43% | 21 | 15 | 15 | 0 | 6 |
| 12475.jpg | **54.55%** | 66.67% | 46.15% | 16 | 9 | 6 | 3 | 7 |
| 12476.jpg | **76.19%** | 88.89% | 66.67% | 13 | 9 | 8 | 1 | 4 |
| 12478.jpg | **58.33%** | 77.78% | 46.67% | 17 | 9 | 7 | 2 | 8 |
| 12479.jpg | **53.33%** | 57.14% | 50.00% | 11 | 7 | 4 | 3 | 4 |
| 12480.jpg | **68.09%** | 76.19% | 61.54% | 31 | 21 | 16 | 5 | 10 |
| 12481.jpg | **64.00%** | 80.00% | 53.33% | 17 | 10 | 8 | 2 | 7 |
| 12483.jpg | **60.87%** | 63.64% | 58.33% | 16 | 11 | 7 | 4 | 5 |
| 12489.jpg | **75.00%** | 81.82% | 69.23% | 15 | 11 | 9 | 2 | 4 |
| 12490.jpg | **72.73%** | 80.00% | 66.67% | 21 | 15 | 12 | 3 | 6 |
| 12492.jpg | **77.27%** | 80.95% | 73.91% | 27 | 21 | 17 | 4 | 6 |
| 12495.jpg | **62.50%** | 71.43% | 55.56% | 11 | 7 | 5 | 2 | 4 |
| 12497.jpg | **83.33%** | 78.95% | 88.24% | 21 | 19 | 15 | 4 | 2 |
| 12500.jpg | **77.42%** | 100.00% | 63.16% | 19 | 12 | 12 | 0 | 7 |
| 12503.jpg | **53.33%** | 66.67% | 44.44% | 22 | 12 | 8 | 4 | 10 |
| 12505.jpg | **66.67%** | 76.92% | 58.82% | 20 | 13 | 10 | 3 | 7 |
| 12507.jpg | **83.72%** | 90.00% | 78.26% | 25 | 20 | 18 | 2 | 5 |

**Table 4.11** Independent Sample T-Test Result for Translucent Colonies Only

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Actual | CHT (Kis) | CHT (Alternative) | BD (Alternative) |
| Mean | 18.53333 | 0 | 14.9 | 12.13333 |
| Size | 30 | 30 | 30 | 30 |
| Variance | 372.6667 | 0 | 248.6667 | 159.9333 |
| df |  | 58 | 58 | 58 |
| Critical Value |  | 1.671 | 1.671 | 1.671 |
| SEDM |  | 3.58477 | 4.628746 | 4.285501 |
| t-statistic |  | **5.17002** | **0.78495** | **1.493408** |

For the independent sample t-test for the Translucent Colony count, only the t-statistic of Circular Hough Transform used by Kis were greater than the critical value of 1.671, stating that there are significant difference between the counted colony using Kis’ process and the actual colony count. On the other hand, the t-statistic of the Circular Hough Transform with Alternative Process and Blob Detection were less than the critical value of 1.671. Which means that there is no significant difference between the number of colonies that the counting process provided and the actual number of colonies in the image.

**Table 4.12** Combined Colony Count Results for the Method of Kis et al. [9]

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID | F-Score | Precision | Recall | Actual | Counted | TP | FP | FN |
| 12442.jpg | **73.33%** | 89.19% | 62.26% | 57 | 37 | 33 | 4 | 20 |
| 12444.jpg | **33.96%** | 90.00% | 20.93% | 44 | 10 | 9 | 1 | 34 |
| 12452.jpg | **45.83%** | 91.67% | 30.56% | 37 | 12 | 11 | 1 | 25 |
| 12454.jpg | **80.00%** | 72.73% | 88.89% | 11 | 11 | 8 | 3 | 1 |
| 12455.jpg | **57.58%** | 86.36% | 43.18% | 47 | 22 | 19 | 3 | 25 |
| 12456.jpg | **33.33%** | 25.00% | 50.00% | 11 | 20 | 5 | 15 | 5 |
| 12457.jpg | **32.35%** | 100.00% | 19.30% | 57 | 11 | 11 | 0 | 46 |
| 12460.jpg | **35.00%** | 63.64% | 24.14% | 33 | 11 | 7 | 4 | 22 |
| 12461.jpg | **20.41%** | 71.43% | 11.90% | 44 | 7 | 5 | 2 | 37 |
| 12463.jpg | **42.11%** | 42.86% | 41.38% | 38 | 28 | 12 | 16 | 17 |
| 12465.jpg | **36.07%** | 45.83% | 29.73% | 50 | 24 | 11 | 13 | 26 |
| 12466.jpg | **59.65%** | 77.27% | 48.57% | 40 | 22 | 17 | 5 | 18 |
| 12470.jpg | **36.36%** | 100.00% | 22.22% | 45 | 10 | 10 | 0 | 35 |
| 12471.jpg | **0.00%** | 0.00% | 0.00% | 50 | 2 | 0 | 2 | 48 |
| 12475.jpg | **48.54%** | 60.98% | 40.32% | 78 | 41 | 25 | 16 | 37 |
| 12476.jpg | **62.30%** | 65.52% | 59.38% | 37 | 29 | 19 | 10 | 13 |
| 12478.jpg | **41.86%** | 75.00% | 29.03% | 34 | 12 | 9 | 3 | 22 |
| 12479.jpg | **86.87%** | 97.73% | 78.18% | 56 | 44 | 43 | 1 | 12 |
| 12480.jpg | **14.55%** | 80.00% | 8.00% | 51 | 5 | 4 | 1 | 46 |
| 12481.jpg | **42.86%** | 100.00% | 27.27% | 44 | 12 | 12 | 0 | 32 |
| 12483.jpg | **41.94%** | 70.27% | 29.89% | 98 | 37 | 26 | 11 | 61 |
| 12489.jpg | **32.56%** | 33.33% | 31.82% | 33 | 21 | 7 | 14 | 15 |
| 12490.jpg | **60.00%** | 75.00% | 50.00% | 46 | 28 | 21 | 7 | 21 |
| 12492.jpg | **17.78%** | 28.57% | 12.90% | 37 | 14 | 4 | 10 | 27 |
| 12495.jpg | **73.33%** | 81.48% | 66.67% | 37 | 27 | 22 | 5 | 11 |
| 12497.jpg | **58.43%** | 76.47% | 47.27% | 63 | 34 | 26 | 8 | 29 |
| 12500.jpg | **16.00%** | 100.00% | 8.70% | 69 | 6 | 6 | 0 | 63 |
| 12503.jpg | **61.76%** | 87.50% | 47.73% | 47 | 24 | 21 | 3 | 23 |
| 12505.jpg | **52.31%** | 60.71% | 45.95% | 46 | 28 | 17 | 11 | 20 |
| 12507.jpg | **17.02%** | 100.00% | 9.30% | 43 | 4 | 4 | 0 | 39 |

**Table 4.13** Combined Colony Count Results for the Alternative Process using Circular Hough Transform

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID | F-Score | Precision | Recall | Actual | Counted | TP | FP | FN |
| 12442.jpg | **64.29%** | 60.00% | 69.23% | 57 | 45 | 27 | 18 | 12 |
| 12444.jpg | **30.77%** | 22.86% | 47.06% | 44 | 35 | 8 | 27 | 9 |
| 12452.jpg | **56.60%** | 41.67% | 88.24% | 37 | 36 | 15 | 21 | 2 |
| 12454.jpg | **55.56%** | 41.67% | 83.33% | 11 | 12 | 5 | 7 | 1 |
| 12455.jpg | **57.97%** | 47.62% | 74.07% | 47 | 42 | 20 | 22 | 7 |
| 12456.jpg | **25.00%** | 17.39% | 44.44% | 11 | 23 | 4 | 19 | 5 |
| 12457.jpg | **65.88%** | 60.87% | 71.79% | 57 | 46 | 28 | 18 | 11 |
| 12460.jpg | **52.63%** | 45.45% | 62.50% | 33 | 33 | 15 | 18 | 9 |
| 12461.jpg | **57.14%** | 47.37% | 72.00% | 44 | 38 | 18 | 20 | 7 |
| 12463.jpg | **51.85%** | 40.00% | 73.68% | 38 | 35 | 14 | 21 | 5 |
| 12465.jpg | **56.00%** | 50.00% | 63.64% | 50 | 42 | 21 | 21 | 12 |
| 12466.jpg | **57.14%** | 44.44% | 80.00% | 40 | 36 | 16 | 20 | 4 |
| 12470.jpg | **69.57%** | 58.54% | 85.71% | 45 | 41 | 24 | 17 | 4 |
| 12471.jpg | **47.76%** | 42.11% | 55.17% | 50 | 38 | 16 | 22 | 13 |
| 12475.jpg | **48.54%** | 60.98% | 40.32% | 78 | 41 | 25 | 16 | 37 |
| 12476.jpg | **66.67%** | 58.82% | 76.92% | 37 | 34 | 20 | 14 | 6 |
| 12478.jpg | **44.00%** | 32.35% | 68.75% | 34 | 34 | 11 | 23 | 5 |
| 12479.jpg | **67.42%** | 62.50% | 73.17% | 56 | 48 | 30 | 18 | 11 |
| 12480.jpg | **58.33%** | 56.76% | 60.00% | 51 | 37 | 21 | 16 | 14 |
| 12481.jpg | **66.67%** | 57.89% | 78.57% | 44 | 38 | 22 | 16 | 6 |
| 12483.jpg | **49.23%** | 62.75% | 40.51% | 98 | 51 | 32 | 19 | 47 |
| 12489.jpg | **50.00%** | 36.36% | 80.00% | 33 | 33 | 12 | 21 | 3 |
| 12490.jpg | **60.61%** | 48.78% | 80.00% | 46 | 41 | 20 | 21 | 5 |
| 12492.jpg | **70.18%** | 57.14% | 90.91% | 37 | 35 | 20 | 15 | 2 |
| 12495.jpg | **73.02%** | 57.50% | 100.00% | 37 | 40 | 23 | 17 | 0 |
| 12497.jpg | **65.96%** | 72.09% | 60.78% | 63 | 43 | 31 | 12 | 20 |
| 12500.jpg | **23.08%** | 23.68% | 22.50% | 69 | 38 | 9 | 29 | 31 |
| 12503.jpg | **70.73%** | 67.44% | 74.36% | 47 | 43 | 29 | 14 | 10 |
| 12505.jpg | **73.97%** | 62.79% | 90.00% | 46 | 43 | 27 | 16 | 3 |
| 12507.jpg | **65.63%** | 53.85% | 84.00% | 43 | 39 | 21 | 18 | 4 |

**Table 4.14** Combined Colony Count for the Alternative Process using Blob Detection

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID | F-Score | Precision | Recall | Actual | Counted | TP | FP | FN |
| 12442.jpg | **83.67%** | 97.62% | 73.21% | 57 | 42 | 41 | 1 | 15 |
| 12444.jpg | **72.46%** | 89.29% | 60.98% | 44 | 28 | 25 | 3 | 16 |
| 12452.jpg | **74.58%** | 84.62% | 66.67% | 37 | 26 | 22 | 4 | 11 |
| 12454.jpg | **90.00%** | 100.00% | 81.82% | 11 | 9 | 9 | 0 | 2 |
| 12455.jpg | **82.50%** | 89.19% | 76.74% | 47 | 37 | 33 | 4 | 10 |
| 12456.jpg | **62.50%** | 83.33% | 50.00% | 11 | 6 | 5 | 1 | 5 |
| 12457.jpg | **74.73%** | 87.18% | 65.38% | 57 | 39 | 34 | 5 | 18 |
| 12460.jpg | **80.00%** | 88.00% | 73.33% | 33 | 25 | 22 | 3 | 8 |
| 12461.jpg | **72.46%** | 86.21% | 62.50% | 44 | 29 | 25 | 4 | 15 |
| 12463.jpg | **73.33%** | 91.67% | 61.11% | 38 | 24 | 22 | 2 | 14 |
| 12465.jpg | **80.95%** | 87.18% | 75.56% | 50 | 39 | 34 | 5 | 11 |
| 12466.jpg | **78.79%** | 92.86% | 68.42% | 40 | 28 | 26 | 2 | 12 |
| 12470.jpg | **76.71%** | 100.00% | 62.22% | 45 | 28 | 28 | 0 | 17 |
| 12471.jpg | **70.13%** | 96.43% | 55.10% | 50 | 28 | 27 | 1 | 22 |
| 12475.jpg | **66.67%** | 88.64% | 53.42% | 78 | 44 | 39 | 5 | 34 |
| 12476.jpg | **86.15%** | 93.33% | 80.00% | 37 | 30 | 28 | 2 | 7 |
| 12478.jpg | **71.70%** | 79.17% | 65.52% | 34 | 24 | 19 | 5 | 10 |
| 12479.jpg | **83.33%** | 93.02% | 75.47% | 56 | 43 | 40 | 3 | 13 |
| 12480.jpg | **70.89%** | 82.35% | 62.22% | 51 | 34 | 28 | 6 | 17 |
| 12481.jpg | **79.45%** | 93.55% | 69.05% | 44 | 31 | 29 | 2 | 13 |
| 12483.jpg | **71.05%** | 91.53% | 58.06% | 98 | 59 | 54 | 5 | 39 |
| 12489.jpg | **77.78%** | 91.30% | 67.74% | 33 | 23 | 21 | 2 | 10 |
| 12490.jpg | **77.33%** | 90.63% | 67.44% | 46 | 32 | 29 | 3 | 14 |
| 12492.jpg | **76.67%** | 85.19% | 69.70% | 37 | 27 | 23 | 4 | 10 |
| 12495.jpg | **82.54%** | 86.67% | 78.79% | 37 | 30 | 26 | 4 | 7 |
| 12497.jpg | **84.40%** | 88.46% | 80.70% | 63 | 52 | 46 | 6 | 11 |
| 12500.jpg | **68.57%** | 97.30% | 52.94% | 69 | 37 | 36 | 1 | 32 |
| 12503.jpg | **81.01%** | 88.89% | 74.42% | 47 | 36 | 32 | 4 | 11 |
| 12505.jpg | **75.68%** | 90.32% | 65.12% | 46 | 31 | 28 | 3 | 15 |
| 12507.jpg | **82.19%** | 93.75% | 73.17% | 43 | 32 | 30 | 2 | 11 |

**Table 4.15** Independent Sample T-Test Results for Both Opaque and Translucent Colonies

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | ACTUAL | CHT (Kis) | CHT (Alternative) | BLOB (Alternative) |
| Mean | 46.1 | 19.76667 | 38 | 31.76667 |
| Size | 30 | 30 | 30 | 30 |
| Variance | 2356.6 | 504.0667 | 1457.333 | 1086.267 |
| df |  | 58 | 58 | 58 |
| Critical Value |  | 1.671 | 1.671 | 1.671 |
| SEDM |  | 9.931952 | 11.468 | 10.89585 |
| t-statistic |  | **2.651375** | **0.706313** | **1.315486** |

For the independent sample t-test for the combined colony count, only the t-statistic of Circular Hough Transform used by Kis were greater than the critical value of 1.671, stating that there is significant difference between the counted colony using Kis’ process and the actual colony count. On the other hand, the t-statistic of the Circular Hough Transform with Alternative Process and Blob Detection were less than the critical value of 1.671, meaning that there is no significant difference between the number of colonies that the counting process provided and the actual number of colonies in the image.

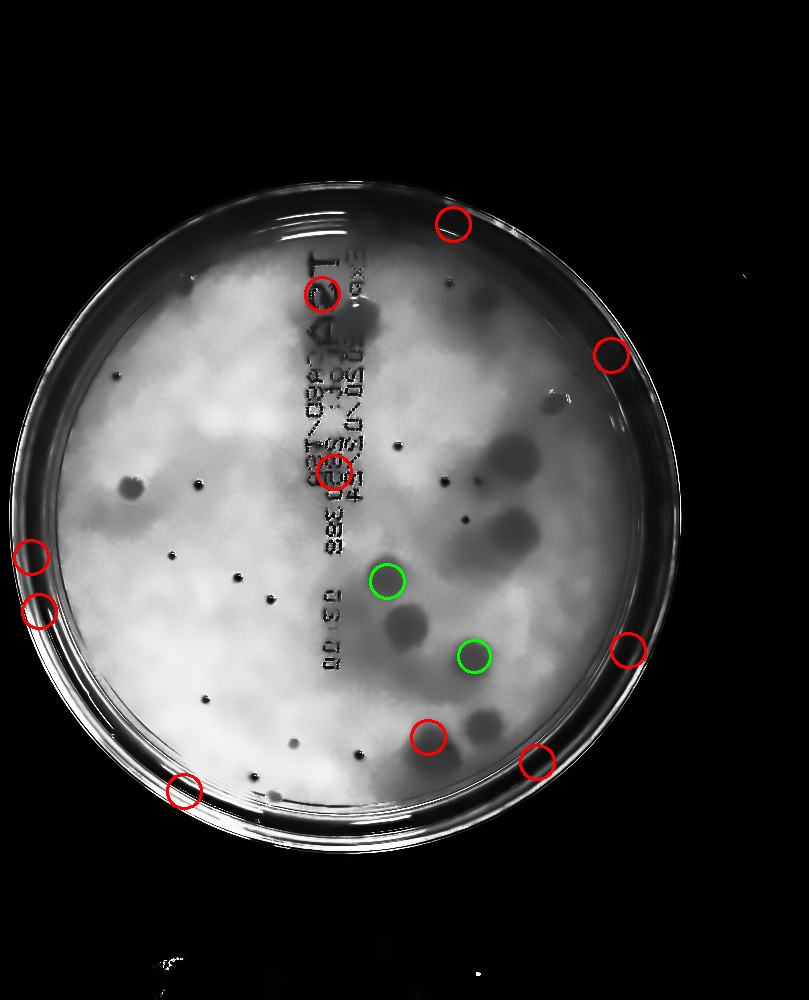
**Table 4.16** Average Performance Overview

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | CHT (Kis) | CHT (Alternative) | BD (Alternative) |
| Opaque | F-Score | 63.05% | 56.61% | 81.58% |
| Precision | 71.62% | 48.79% | 94.63% |
| Recall | 69.24% | 76.76% | 73.37% |
| Translucent | F-Score | 0.00% | 46.59% | 69.18% |
| Precision | 0.00% | 41.22% | 81.63% |
| Recall | 0.00% | 59.90% | 61.53% |
| Combined | F-Score | 43.80% | 56.74% | 76.94% |
| Precision | 71.62% | 49.72% | 90.25% |
| Recall | 36.18% | 69.72% | 67.56% |

Overall, the alternative process using the Blob Detection algorithm consistently achieved the highest F-score in all three categories–81.58% for the opaque colonies, 69.18% for the translucent colonies and a combined 76.94%. However, the other metrics also have some value in measuring performance given certain conditions since the F-score equally weighs the importance of precision and recall. For example, if only the count of the colonies were important, the alternative process using the Circular Hough Transform would be the better choice since it has consistently scored the lowest absolute t-statistic for all three categories despite being less precise than Blob Detection.

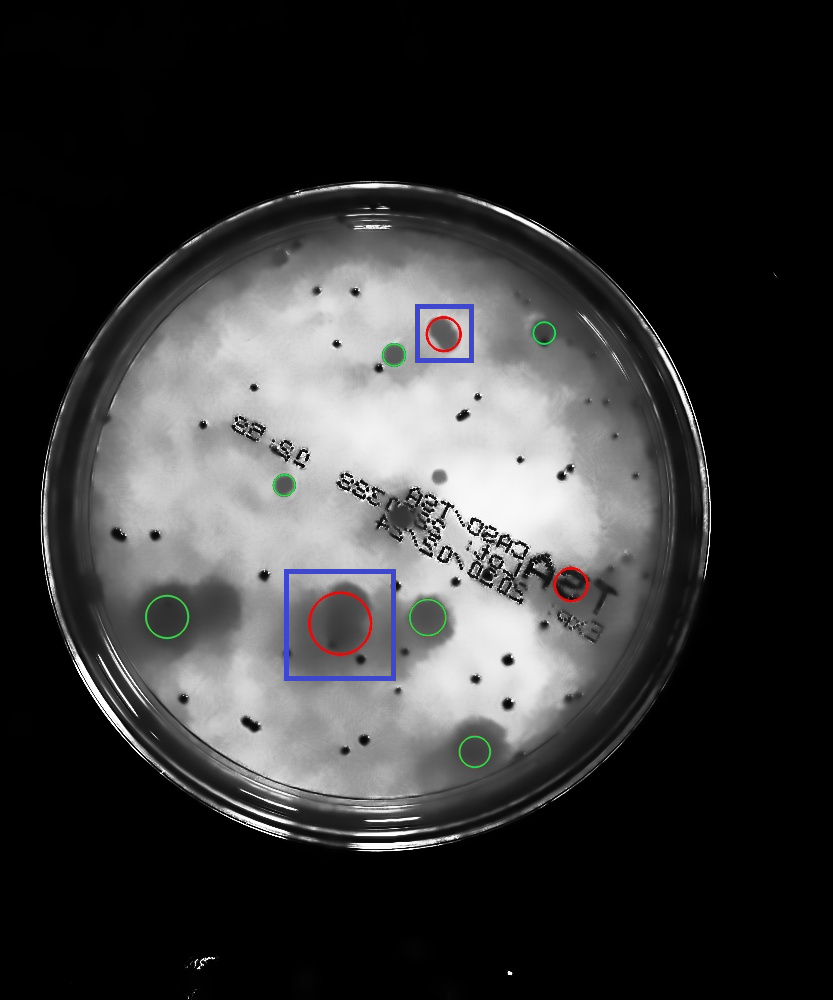
**Error Analysis**

Each of the three implemented methods have some false positive and false negative detections which may be explained based on how the algorithms used work. The appearance of false positives, often on the text and edges of the petri dish as shown in Figure 4.13, which were detected by the methods using the Circular Hough Transform can be explained by the tradeoff of its low accumulator parameter, which can be exceeded with enough temporary circles intersecting, as discussed in the methodology, to confirm that a circle is in the intersection even if there is no colony present in the location.

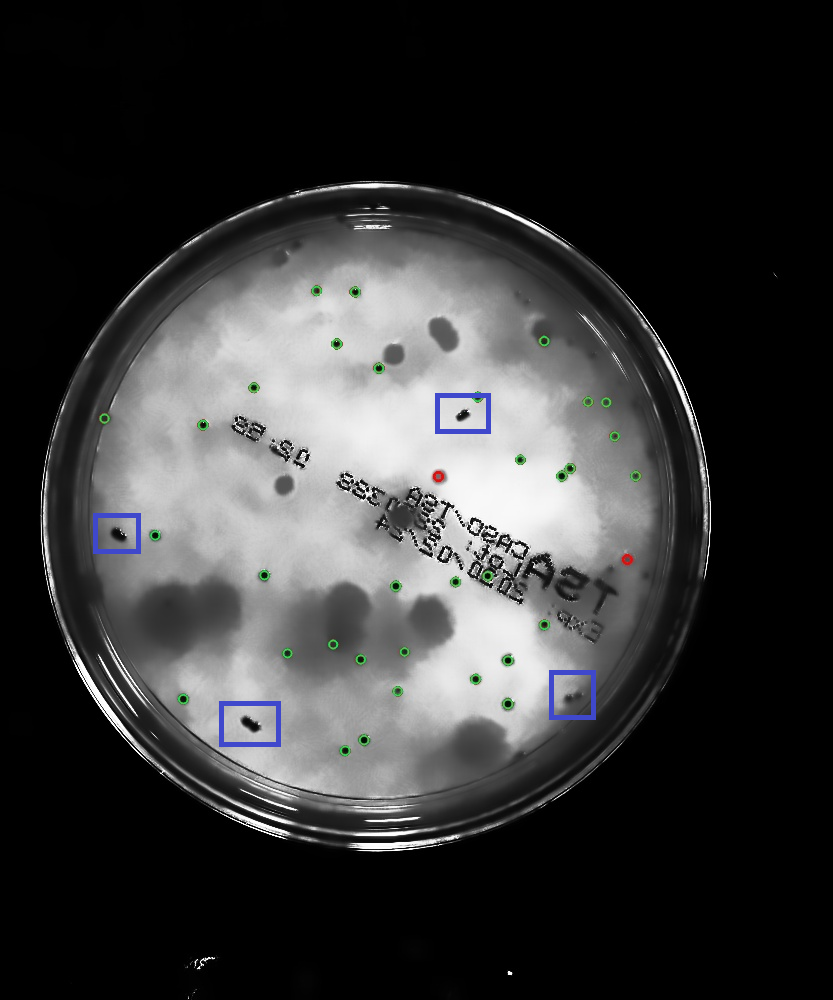


**Figure 4.13** False Positives caused by the Circular Hough Transform (Translucent)

Focusing next on the Blob Detection algorithm, its fundamental flaw is the inability to discern overlapping colonies due to its nature of grouping together pixels of the same intensity. Depending on how close the overlap is between two colonies, it may be detected as one colony due to its convexity threshold or not detected at all. Images 12475 and 12478 in particular have overlapping translucent colonies treated as one, and images 12442, 12444, 12457, 12475, and 12483 have overlapping colonies which are not detected.

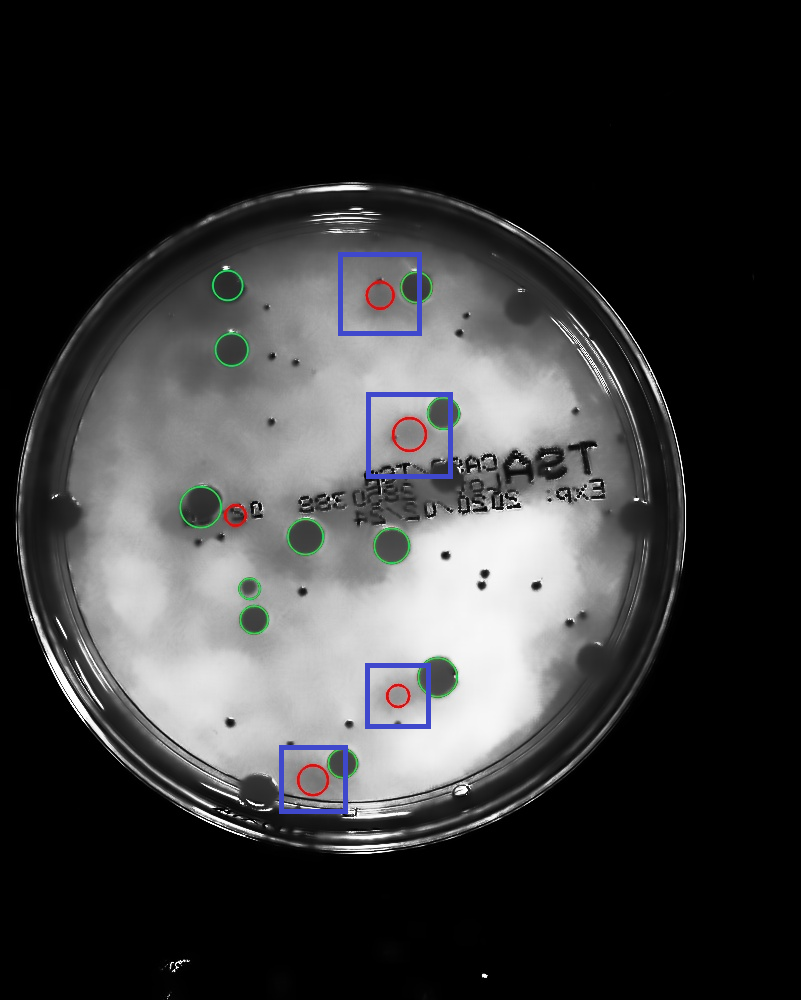


**Figure 4.14** Overlapping Colonies Treated as One by Blob Detection (Translucent)

****

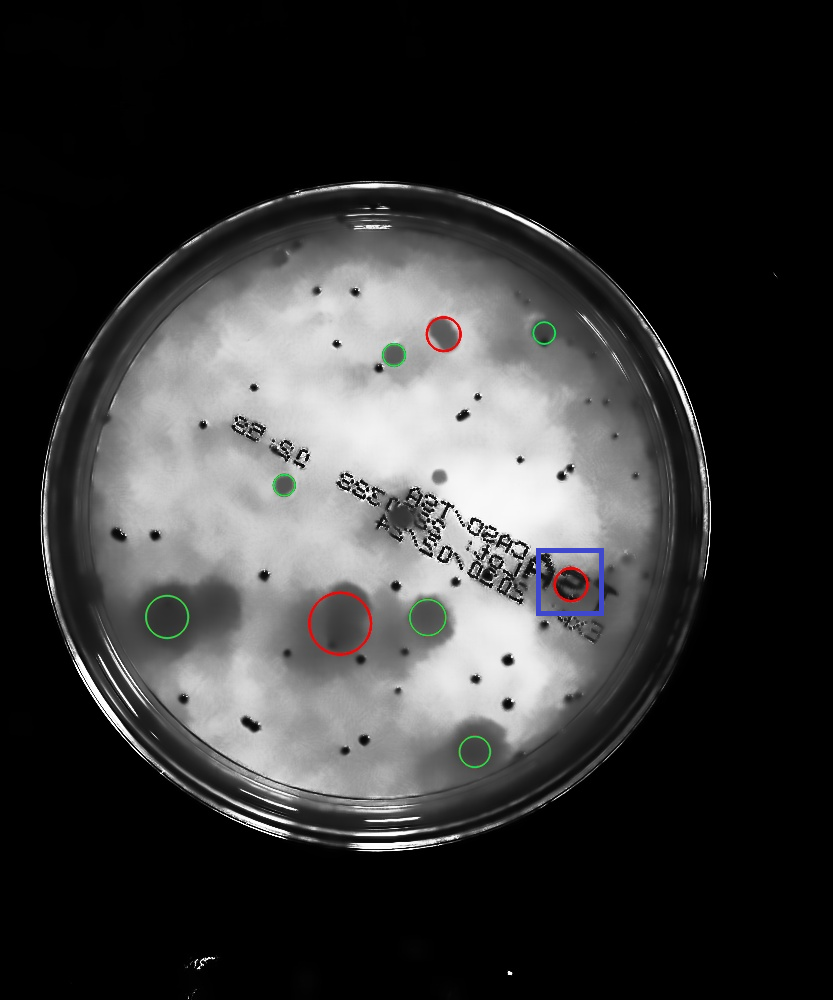
**Figure 4.15** Overlapping Colonies that are not Detected by Blob Detection (Opaque)

Like the Circular Hough Transform, the Blob Detection algorithm also has a tradeoff with its minimum repeatability parameter. The images produced after histogram equalization is applied show areas wherein there is a slightly darker area wherein there is no colony as shown in Figure 4.16. Due to having a low minimum repeatability parameter, this area can persist across few iterations of thresholding, potentially exceeding the minimum repeatability and being deemed as a valid blob by the algorithm.



**Figure 4.16** False Positives caused by Histogram Equalization and Blob Detection (Translucent)

There is also a unique error pattern that has been observed which was caused by the Blob Detection algorithm. Images 12460, 12475, 12479, 12483, 12489, and 12495 have the characters ‘S’ and ‘E’ often detected as a false positive by the algorithm. This can be explained by the minimum convexity parameter of the algorithm, which marks blobs above the set value as valid.



**Figure 4.17** Letter ‘S’ Detected by Blob Detection

Aside from the downsides and tradeoffs introduced by the different algorithms, a common occurrence that was found was the inability to detect some colonies near the edges of the dish, which is caused by the shadow coming from the lid.

****

**Figure 4.18** False Negatives Caused by Shadows Near the Edges

**Chapter 5**

# CONCLUSIONS AND RECOMMENDATIONS

## Conclusions

Although automated counting methods were already widely used, the researchers aimed to create a model for bacteria colony detection that can reliably detect and count translucent colonies on a petri dish image. Precisely, to determine algorithms that can enhance colony appearance prior to detection and algorithms for detection itself. Afterwards, combinations of these algorithms were implemented and tested on a sample set based on different metrics. The three proposed methods, namely the Circular Hough Transform used by Kis, Circular Hough Transform with Alternative Preprocessing, and Blob Detection with Alternative Preprocessing, were measured based on based on the precision, recall, and F-score metrics. The Circular Hough Transform method used by Kis scored 71.62% base in precision, 36.18% on recall and 43.80% on F-score. The Circular Hough Transform with alternative preprocessing scored a precision of 49.72%, a recall score of 69.72%, and an F-score of 56.74%. On the other hand, the proposed Blob Detection method with alternative preprocessing performed best based on the highest precision, slightly lower recall, and the highest F-score metrics of 90.25%, 67.56%, and 76.94%, respectively.

Based on a t-test statistical analysis, the study found that a modified implementation of the methodology used by Kis and using Circular Hough Transform achieved colony counts that are closest to the actual count. Overall, both methods using alternative preprocessing have significant improvements in detecting translucent colonies, and both have their strengths and weaknesses.

The appearance of false positives on the text and shadows in between colonies, as well as false negatives near the edges of the petri dish which have been amplified by histogram equalization, and the incorrect detection of overlapping colonies as one colony by the Blob Detection algorithm, were some of the identified errors which affected the resulting performance metrics of each method.

## Recommendations

A method for reducing obstructions from shadows caused by uneven lighting on colonies could be explored by separating the region near the edges of the dish from the rest when performing histogram equalization to improve detection of colonies near the edges. Removal of obstructing text could also be possible by simple thresholding. However, doing so follows that colonies which overlap with the text may also be lost. Being able to retain this information would be useful for the accuracy of detection. On the other hand, the issues that the Blob Detection algorithm has with overlapping colonies could be tackled by detecting blobs within already detected blobs to ensure that two colonies were not merged.

On top of solving the aforementioned issues from the error analysis portion, further studies could also be used to explore additional or alternative ways not limited to the methods employed in this study in improving the detection rate of colonies.

**APPENDIX**

Parameter Testing for Opaque Colony using the Process of Kis et al. [9]

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| F-Score | Precision | Recall | Actual | Count | TP | FP | FN | MN\_DIST | ACCUM | MN\_RAD | MX\_RAD |
| 34.36% | 24.13% | 90.41% | 827 | 2165 | 440 | 1725 | 74 | 1 | 1 | 1 | 1 |
| 34.36% | 24.13% | 90.41% | 827 | 2165 | 440 | 1725 | 74 | 1 | 1 | 1 | 3 |
| 24.84% | 14.96% | 93.98% | 827 | 3291 | 443 | 2848 | 40 | 1 | 1 | 1 | 5 |
| 15.62% | 8.73% | 94.58% | 827 | 5560 | 443 | 5117 | 14 | 1 | 1 | 1 | 7 |
| 33.19% | 21.17% | 93.31% | 827 | 2225 | 442 | 1783 | 48 | 1 | 1 | 3 | 3 |
| 33.19% | 21.17% | 93.31% | 827 | 2225 | 442 | 1783 | 48 | 1 | 1 | 3 | 5 |
| 19.30% | 11.17% | 94.10% | 827 | 4218 | 443 | 3775 | 29 | 1 | 1 | 3 | 7 |
| 15.69% | 8.90% | 93.57% | 827 | 3000 | 302 | 2698 | 41 | 1 | 1 | 5 | 5 |
| 15.69% | 8.90% | 93.57% | 827 | 3000 | 302 | 2698 | 41 | 1 | 1 | 5 | 7 |
| 47.91% | 45.55% | 79.42% | 827 | 1219 | 439 | 780 | 171 | 1 | 2 | 1 | 1 |
| 47.91% | 45.55% | 79.42% | 827 | 1219 | 439 | 780 | 171 | 1 | 2 | 1 | 3 |
| 41.92% | 31.87% | 84.59% | 827 | 1606 | 440 | 1166 | 104 | 1 | 2 | 1 | 5 |
| 30.96% | 20.31% | 90.41% | 827 | 2522 | 441 | 2081 | 65 | 1 | 2 | 1 | 7 |
| 39.60% | 46.79% | 58.41% | 827 | 747 | 298 | 449 | 300 | 1 | 2 | 3 | 3 |
| 39.60% | 46.79% | 58.41% | 827 | 747 | 298 | 449 | 300 | 1 | 2 | 3 | 5 |
| 30.50% | 21.11% | 81.49% | 827 | 1566 | 333 | 1233 | 136 | 1 | 2 | 3 | 7 |
| 29.57% | 20.74% | 68.80% | 827 | 1060 | 265 | 795 | 207 | 1 | 2 | 5 | 5 |
| 29.57% | 20.74% | 68.80% | 827 | 1060 | 265 | 795 | 207 | 1 | 2 | 5 | 7 |
| 51.69% | 51.39% | 77.66% | 827 | 1008 | 435 | 573 | 192 | 1 | 3 | 1 | 1 |
| 51.69% | 51.39% | 77.66% | 827 | 1008 | 435 | 573 | 192 | 1 | 3 | 1 | 3 |
| 49.80% | 47.14% | 79.75% | 827 | 1104 | 438 | 666 | 165 | 1 | 3 | 1 | 5 |
| 43.55% | 38.95% | 82.34% | 827 | 1432 | 438 | 994 | 138 | 1 | 3 | 1 | 7 |
| 44.60% | 58.09% | 50.29% | 827 | 511 | 295 | 216 | 365 | 1 | 3 | 3 | 3 |
| 44.60% | 58.09% | 50.29% | 827 | 511 | 295 | 216 | 365 | 1 | 3 | 3 | 5 |
| 40.17% | 44.65% | 59.82% | 827 | 706 | 295 | 411 | 298 | 1 | 3 | 3 | 7 |
| 22.47% | 30.97% | 26.12% | 827 | 293 | 132 | 161 | 540 | 1 | 3 | 5 | 5 |
| 22.47% | 30.97% | 26.12% | 827 | 293 | 132 | 161 | 540 | 1 | 3 | 5 | 7 |
| 54.43% | 56.07% | 74.55% | 827 | 807 | 405 | 402 | 219 | 1 | 4 | 1 | 1 |
| 54.43% | 56.07% | 74.55% | 827 | 807 | 405 | 402 | 219 | 1 | 4 | 1 | 3 |
| 55.41% | 56.52% | 76.37% | 827 | 873 | 436 | 437 | 196 | 1 | 4 | 1 | 5 |
| 50.19% | 49.64% | 78.50% | 827 | 1066 | 437 | 629 | 180 | 1 | 4 | 1 | 7 |
| 46.40% | 66.05% | 45.45% | 827 | 412 | 292 | 120 | 426 | 1 | 4 | 3 | 3 |
| 46.40% | 66.05% | 45.45% | 827 | 412 | 292 | 120 | 426 | 1 | 4 | 3 | 5 |
| 43.79% | 57.78% | 48.86% | 827 | 510 | 292 | 218 | 383 | 1 | 4 | 3 | 7 |
| 12.49% | 26.89% | 9.66% | 827 | 106 | 69 | 37 | 721 | 1 | 4 | 5 | 5 |
| 12.49% | 26.89% | 9.66% | 827 | 106 | 69 | 37 | 721 | 1 | 4 | 5 | 7 |
| 54.81% | 60.04% | 68.58% | 827 | 638 | 353 | 285 | 269 | 1 | 5 | 1 | 1 |
| 54.81% | 60.04% | 68.58% | 827 | 638 | 353 | 285 | 269 | 1 | 5 | 1 | 3 |
| 58.97% | 61.64% | 74.23% | 827 | 752 | 434 | 318 | 223 | 1 | 5 | 1 | 5 |
| 53.34% | 53.79% | 77.53% | 827 | 940 | 436 | 504 | 193 | 1 | 5 | 1 | 7 |
| 42.61% | 64.53% | 37.49% | 827 | 323 | 259 | 64 | 505 | 1 | 5 | 3 | 3 |
| 42.61% | 64.53% | 37.49% | 827 | 323 | 259 | 64 | 505 | 1 | 5 | 3 | 5 |
| 41.89% | 59.17% | 39.98% | 827 | 371 | 261 | 110 | 472 | 1 | 5 | 3 | 7 |
| 6.00% | 24.38% | 4.07% | 827 | 40 | 31 | 9 | 787 | 1 | 5 | 5 | 5 |
| 6.00% | 24.38% | 4.07% | 827 | 40 | 31 | 9 | 787 | 1 | 5 | 5 | 7 |
| 51.60% | 61.76% | 59.20% | 827 | 522 | 304 | 218 | 347 | 1 | 6 | 1 | 1 |
| 51.60% | 61.76% | 59.20% | 827 | 522 | 304 | 218 | 347 | 1 | 6 | 1 | 3 |
| 60.65% | 65.73% | 71.67% | 827 | 670 | 426 | 244 | 248 | 1 | 6 | 1 | 5 |
| 57.20% | 60.03% | 74.08% | 827 | 783 | 431 | 352 | 220 | 1 | 6 | 1 | 7 |
| 37.28% | 73.16% | 29.94% | 827 | 249 | 223 | 26 | 578 | 1 | 6 | 3 | 3 |
| 37.28% | 73.16% | 29.94% | 827 | 249 | 223 | 26 | 578 | 1 | 6 | 3 | 5 |
| 37.36% | 69.13% | 31.99% | 827 | 274 | 227 | 47 | 555 | 1 | 6 | 3 | 7 |
| 5.19% | 18.12% | 3.60% | 827 | 29 | 26 | 3 | 798 | 1 | 6 | 5 | 5 |
| 5.19% | 18.12% | 3.60% | 827 | 29 | 26 | 3 | 798 | 1 | 6 | 5 | 7 |
| 46.84% | 61.99% | 49.52% | 827 | 436 | 260 | 176 | 416 | 1 | 7 | 1 | 1 |
| 46.84% | 61.99% | 49.52% | 827 | 436 | 260 | 176 | 416 | 1 | 7 | 1 | 3 |
| 61.91% | 69.53% | 69.42% | 827 | 583 | 409 | 174 | 279 | 1 | 7 | 1 | 5 |
| 60.15% | 64.85% | 72.11% | 827 | 666 | 423 | 243 | 250 | 1 | 7 | 1 | 7 |
| 34.70% | 76.24% | 25.37% | 827 | 211 | 201 | 10 | 616 | 1 | 7 | 3 | 3 |
| 34.70% | 76.24% | 25.37% | 827 | 211 | 201 | 10 | 616 | 1 | 7 | 3 | 5 |
| 35.72% | 74.83% | 27.85% | 827 | 233 | 212 | 21 | 594 | 1 | 7 | 3 | 7 |
| 3.36% | 13.10% | 2.39% | 827 | 22 | 20 | 2 | 805 | 1 | 7 | 5 | 5 |
| 3.36% | 13.10% | 2.39% | 827 | 22 | 20 | 2 | 805 | 1 | 7 | 5 | 7 |
| 42.93% | 32.35% | 88.17% | 827 | 1516 | 437 | 1079 | 89 | 3 | 1 | 1 | 1 |
| 42.93% | 32.35% | 88.17% | 827 | 1516 | 437 | 1079 | 89 | 3 | 1 | 1 | 3 |
| 41.29% | 30.10% | 88.95% | 827 | 1664 | 435 | 1229 | 80 | 3 | 1 | 1 | 5 |
| 27.71% | 17.75% | 93.81% | 827 | 2775 | 437 | 2338 | 52 | 3 | 1 | 1 | 7 |
| 46.57% | 37.58% | 82.82% | 827 | 1232 | 417 | 815 | 114 | 3 | 1 | 3 | 3 |
| 46.57% | 37.58% | 82.82% | 827 | 1232 | 417 | 815 | 114 | 3 | 1 | 3 | 5 |
| 28.04% | 17.47% | 93.94% | 827 | 2444 | 409 | 2035 | 42 | 3 | 1 | 3 | 7 |
| 17.23% | 9.99% | 93.57% | 827 | 2364 | 263 | 2101 | 43 | 3 | 1 | 5 | 5 |
| 17.23% | 9.99% | 93.57% | 827 | 2364 | 263 | 2101 | 43 | 3 | 1 | 5 | 7 |
| 55.25% | 56.29% | 74.97% | 827 | 872 | 433 | 439 | 209 | 3 | 2 | 1 | 1 |
| 55.25% | 56.29% | 74.97% | 827 | 872 | 433 | 439 | 209 | 3 | 2 | 1 | 3 |
| 55.39% | 55.06% | 76.59% | 827 | 911 | 433 | 478 | 190 | 3 | 2 | 1 | 5 |
| 48.74% | 42.19% | 81.51% | 827 | 1220 | 433 | 787 | 134 | 3 | 2 | 1 | 7 |
| 45.12% | 60.85% | 52.85% | 827 | 517 | 295 | 222 | 365 | 3 | 2 | 3 | 3 |
| 45.12% | 60.85% | 52.85% | 827 | 517 | 295 | 222 | 365 | 3 | 2 | 3 | 5 |
| 40.16% | 33.16% | 72.94% | 827 | 954 | 316 | 638 | 195 | 3 | 2 | 3 | 7 |
| 35.32% | 27.84% | 63.07% | 827 | 769 | 249 | 520 | 244 | 3 | 2 | 5 | 5 |
| 35.32% | 27.84% | 63.07% | 827 | 769 | 249 | 520 | 244 | 3 | 2 | 5 | 7 |
| 58.58% | 63.01% | 72.93% | 827 | 734 | 428 | 306 | 234 | 3 | 3 | 1 | 1 |
| 58.58% | 63.01% | 72.93% | 827 | 734 | 428 | 306 | 234 | 3 | 3 | 1 | 3 |
| 59.76% | 64.89% | 73.13% | 827 | 727 | 435 | 292 | 233 | 3 | 3 | 1 | 5 |
| 57.26% | 59.45% | 74.75% | 827 | 821 | 433 | 388 | 208 | 3 | 3 | 1 | 7 |
| 46.65% | 68.64% | 43.93% | 827 | 399 | 294 | 105 | 434 | 3 | 3 | 3 | 3 |
| 46.65% | 68.64% | 43.93% | 827 | 399 | 294 | 105 | 434 | 3 | 3 | 3 | 5 |
| 45.86% | 58.93% | 49.38% | 827 | 478 | 294 | 184 | 378 | 3 | 3 | 3 | 7 |
| 22.72% | 36.79% | 20.24% | 827 | 243 | 132 | 111 | 585 | 3 | 3 | 5 | 5 |
| 22.72% | 36.79% | 20.24% | 827 | 243 | 132 | 111 | 585 | 3 | 3 | 5 | 7 |
| 58.40% | 65.97% | 69.12% | 827 | 652 | 401 | 251 | 270 | 3 | 4 | 1 | 1 |
| 58.40% | 65.97% | 69.12% | 827 | 652 | 401 | 251 | 270 | 3 | 4 | 1 | 3 |
| 60.96% | 67.43% | 71.96% | 827 | 677 | 434 | 243 | 246 | 3 | 4 | 1 | 5 |
| 60.54% | 66.58% | 72.99% | 827 | 702 | 435 | 267 | 237 | 3 | 4 | 1 | 7 |
| 46.97% | 73.81% | 40.89% | 827 | 354 | 292 | 62 | 475 | 3 | 4 | 3 | 3 |
| 46.97% | 73.81% | 40.89% | 827 | 354 | 292 | 62 | 475 | 3 | 4 | 3 | 5 |
| 46.74% | 70.21% | 42.96% | 827 | 382 | 292 | 90 | 449 | 3 | 4 | 3 | 7 |
| 12.49% | 30.80% | 9.05% | 827 | 94 | 69 | 25 | 733 | 3 | 4 | 5 | 5 |
| 12.49% | 30.80% | 9.05% | 827 | 94 | 69 | 25 | 733 | 3 | 4 | 5 | 7 |
| 55.45% | 65.86% | 62.73% | 827 | 571 | 351 | 220 | 318 | 3 | 5 | 1 | 1 |
| 55.45% | 65.86% | 62.73% | 827 | 571 | 351 | 220 | 318 | 3 | 5 | 1 | 3 |
| 62.12% | 69.48% | 70.89% | 827 | 638 | 432 | 206 | 260 | 3 | 5 | 1 | 5 |
| 61.49% | 68.18% | 71.85% | 827 | 667 | 434 | 233 | 247 | 3 | 5 | 1 | 7 |
| 42.67% | 69.74% | 35.15% | 827 | 292 | 259 | 33 | 535 | 3 | 5 | 3 | 3 |
| 42.67% | 69.74% | 35.15% | 827 | 292 | 259 | 33 | 535 | 3 | 5 | 3 | 5 |
| 42.81% | 67.70% | 35.96% | 827 | 308 | 261 | 47 | 519 | 3 | 5 | 3 | 7 |
| 6.00% | 26.67% | 3.98% | 827 | 37 | 31 | 6 | 790 | 3 | 5 | 5 | 5 |
| 6.00% | 26.67% | 3.98% | 827 | 37 | 31 | 6 | 790 | 3 | 5 | 5 | 7 |
| 51.89% | 65.35% | 56.54% | 827 | 488 | 302 | 186 | 372 | 3 | 6 | 1 | 1 |
| 51.89% | 65.35% | 56.54% | 827 | 488 | 302 | 186 | 372 | 3 | 6 | 1 | 3 |
| 62.98% | 71.52% | 69.32% | 827 | 596 | 425 | 171 | 273 | 3 | 6 | 1 | 5 |
| 62.11% | 69.65% | 70.76% | 827 | 632 | 430 | 202 | 259 | 3 | 6 | 1 | 7 |
| 37.28% | 76.00% | 28.74% | 827 | 234 | 223 | 11 | 593 | 3 | 6 | 3 | 3 |
| 37.28% | 76.00% | 28.74% | 827 | 234 | 223 | 11 | 593 | 3 | 6 | 3 | 5 |
| 37.48% | 73.90% | 29.40% | 827 | 244 | 227 | 17 | 583 | 3 | 6 | 3 | 7 |
| 5.19% | 20.00% | 3.52% | 827 | 27 | 26 | 1 | 800 | 3 | 6 | 5 | 5 |
| 5.19% | 20.00% | 3.52% | 827 | 27 | 26 | 1 | 800 | 3 | 6 | 5 | 7 |
| 46.86% | 64.33% | 47.98% | 827 | 417 | 258 | 159 | 430 | 3 | 7 | 1 | 1 |
| 46.86% | 64.33% | 47.98% | 827 | 417 | 258 | 159 | 430 | 3 | 7 | 1 | 3 |
| 62.71% | 73.43% | 66.26% | 827 | 546 | 408 | 138 | 301 | 3 | 7 | 1 | 5 |
| 62.71% | 71.33% | 69.43% | 827 | 591 | 422 | 169 | 275 | 3 | 7 | 1 | 7 |
| 34.70% | 77.74% | 25.22% | 827 | 207 | 201 | 6 | 620 | 3 | 7 | 3 | 3 |
| 34.70% | 77.74% | 25.22% | 827 | 207 | 201 | 6 | 620 | 3 | 7 | 3 | 5 |
| 35.72% | 77.76% | 26.70% | 827 | 220 | 212 | 8 | 607 | 3 | 7 | 3 | 7 |
| 3.36% | 13.33% | 2.32% | 827 | 21 | 20 | 1 | 806 | 3 | 7 | 5 | 5 |
| 3.36% | 13.33% | 2.32% | 827 | 21 | 20 | 1 | 806 | 3 | 7 | 5 | 7 |
| 47.28% | 37.70% | 85.29% | 827 | 1237 | 418 | 819 | 116 | 5 | 1 | 1 | 1 |
| 47.28% | 37.70% | 85.29% | 827 | 1237 | 418 | 819 | 116 | 5 | 1 | 1 | 3 |
| 46.30% | 36.00% | 85.68% | 827 | 1337 | 433 | 904 | 97 | 5 | 1 | 1 | 5 |
| 40.66% | 29.26% | 91.17% | 827 | 1611 | 425 | 1186 | 71 | 5 | 1 | 1 | 7 |
| 49.88% | 45.42% | 75.91% | 827 | 948 | 390 | 558 | 176 | 5 | 1 | 3 | 3 |
| 49.88% | 45.42% | 75.91% | 827 | 948 | 390 | 558 | 176 | 5 | 1 | 3 | 5 |
| 34.63% | 23.63% | 91.01% | 827 | 1585 | 352 | 1233 | 65 | 5 | 1 | 3 | 7 |
| 13.60% | 8.10% | 67.33% | 827 | 1607 | 145 | 1462 | 70 | 5 | 1 | 5 | 5 |
| 13.60% | 8.10% | 67.33% | 827 | 1607 | 145 | 1462 | 70 | 5 | 1 | 5 | 7 |
| 57.98% | 59.91% | 74.37% | 827 | 774 | 422 | 352 | 220 | 5 | 2 | 1 | 1 |
| 57.98% | 59.91% | 74.37% | 827 | 774 | 422 | 352 | 220 | 5 | 2 | 1 | 3 |
| 57.51% | 58.72% | 75.40% | 827 | 826 | 433 | 393 | 203 | 5 | 2 | 1 | 5 |
| 54.42% | 51.18% | 78.38% | 827 | 940 | 426 | 514 | 169 | 5 | 2 | 1 | 7 |
| 45.41% | 62.22% | 50.13% | 827 | 493 | 294 | 199 | 378 | 5 | 2 | 3 | 3 |
| 45.41% | 62.22% | 50.13% | 827 | 493 | 294 | 199 | 378 | 5 | 2 | 3 | 5 |
| 41.97% | 40.65% | 63.01% | 827 | 684 | 282 | 402 | 275 | 5 | 2 | 3 | 7 |
| 32.36% | 29.73% | 49.05% | 827 | 555 | 194 | 361 | 329 | 5 | 2 | 5 | 5 |
| 32.36% | 29.73% | 49.05% | 827 | 555 | 194 | 361 | 329 | 5 | 2 | 5 | 7 |
| 60.20% | 65.28% | 72.54% | 827 | 686 | 418 | 268 | 240 | 5 | 3 | 1 | 1 |
| 60.20% | 65.28% | 72.54% | 827 | 686 | 418 | 268 | 240 | 5 | 3 | 1 | 3 |
| 60.58% | 66.49% | 73.01% | 827 | 704 | 433 | 271 | 235 | 5 | 3 | 1 | 5 |
| 60.07% | 64.54% | 73.05% | 827 | 723 | 431 | 292 | 228 | 5 | 3 | 1 | 7 |
| 46.71% | 68.95% | 43.77% | 827 | 396 | 294 | 102 | 436 | 5 | 3 | 3 | 3 |
| 46.71% | 68.95% | 43.77% | 827 | 396 | 294 | 102 | 436 | 5 | 3 | 3 | 5 |
| 45.93% | 62.66% | 47.26% | 827 | 449 | 292 | 157 | 397 | 5 | 3 | 3 | 7 |
| 22.62% | 38.72% | 19.10% | 827 | 220 | 131 | 89 | 607 | 5 | 3 | 5 | 5 |
| 22.62% | 38.72% | 19.10% | 827 | 220 | 131 | 89 | 607 | 5 | 3 | 5 | 7 |
| 58.81% | 66.93% | 68.36% | 827 | 632 | 396 | 236 | 276 | 5 | 4 | 1 | 1 |
| 58.81% | 66.93% | 68.36% | 827 | 632 | 396 | 236 | 276 | 5 | 4 | 1 | 3 |
| 61.51% | 68.07% | 71.88% | 827 | 665 | 432 | 233 | 247 | 5 | 4 | 1 | 5 |
| 61.35% | 67.91% | 72.23% | 827 | 676 | 434 | 242 | 244 | 5 | 4 | 1 | 7 |
| 46.97% | 73.93% | 40.75% | 827 | 353 | 292 | 61 | 476 | 5 | 4 | 3 | 3 |
| 46.97% | 73.93% | 40.75% | 827 | 353 | 292 | 61 | 476 | 5 | 4 | 3 | 5 |
| 46.74% | 70.38% | 42.78% | 827 | 381 | 292 | 89 | 450 | 5 | 4 | 3 | 7 |
| 12.49% | 30.97% | 9.03% | 827 | 93 | 69 | 24 | 734 | 5 | 4 | 5 | 5 |
| 12.49% | 30.97% | 9.03% | 827 | 93 | 69 | 24 | 734 | 5 | 4 | 5 | 7 |
| 55.69% | 66.31% | 62.27% | 827 | 560 | 349 | 211 | 322 | 5 | 5 | 1 | 1 |
| 55.69% | 66.31% | 62.27% | 827 | 560 | 349 | 211 | 322 | 5 | 5 | 1 | 3 |
| 62.48% | 69.93% | 70.82% | 827 | 630 | 431 | 199 | 261 | 5 | 5 | 1 | 5 |
| 62.10% | 68.96% | 71.60% | 827 | 651 | 433 | 218 | 250 | 5 | 5 | 1 | 7 |
| 42.67% | 69.74% | 35.15% | 827 | 292 | 259 | 33 | 535 | 5 | 5 | 3 | 3 |
| 42.67% | 69.74% | 35.15% | 827 | 292 | 259 | 33 | 535 | 5 | 5 | 3 | 5 |
| 42.81% | 67.70% | 35.96% | 827 | 308 | 261 | 47 | 519 | 5 | 5 | 3 | 7 |
| 6.00% | 26.67% | 3.98% | 827 | 37 | 31 | 6 | 790 | 5 | 5 | 5 | 5 |
| 6.00% | 26.67% | 3.98% | 827 | 37 | 31 | 6 | 790 | 5 | 5 | 5 | 7 |
| 51.82% | 65.36% | 56.28% | 827 | 486 | 301 | 185 | 374 | 5 | 6 | 1 | 1 |
| 51.82% | 65.36% | 56.28% | 827 | 486 | 301 | 185 | 374 | 5 | 6 | 1 | 3 |
| 63.05% | 71.62% | 69.25% | 827 | 593 | 424 | 169 | 274 | 5 | 6 | 1 | 5 |
| 62.24% | 69.87% | 70.61% | 827 | 626 | 429 | 197 | 261 | 5 | 6 | 1 | 7 |
| 37.28% | 76.00% | 28.74% | 827 | 234 | 223 | 11 | 593 | 5 | 6 | 3 | 3 |
| 37.28% | 76.00% | 28.74% | 827 | 234 | 223 | 11 | 593 | 5 | 6 | 3 | 5 |
| 37.48% | 73.90% | 29.40% | 827 | 244 | 227 | 17 | 583 | 5 | 6 | 3 | 7 |
| 5.19% | 20.00% | 3.52% | 827 | 27 | 26 | 1 | 800 | 5 | 6 | 5 | 5 |
| 5.19% | 20.00% | 3.52% | 827 | 27 | 26 | 1 | 800 | 5 | 6 | 5 | 7 |
| 46.86% | 64.39% | 47.84% | 827 | 416 | 258 | 158 | 431 | 5 | 7 | 1 | 1 |
| 46.86% | 64.39% | 47.84% | 827 | 416 | 258 | 158 | 431 | 5 | 7 | 1 | 3 |
| 62.77% | 73.51% | 66.26% | 827 | 545 | 408 | 137 | 301 | 5 | 7 | 1 | 5 |
| 62.84% | 71.56% | 69.30% | 827 | 586 | 421 | 165 | 277 | 5 | 7 | 1 | 7 |
| 34.70% | 77.74% | 25.22% | 827 | 207 | 201 | 6 | 620 | 5 | 7 | 3 | 3 |
| 34.70% | 77.74% | 25.22% | 827 | 207 | 201 | 6 | 620 | 5 | 7 | 3 | 5 |
| 35.72% | 77.76% | 26.70% | 827 | 220 | 212 | 8 | 607 | 5 | 7 | 3 | 7 |
| 3.36% | 13.33% | 2.32% | 827 | 21 | 20 | 1 | 806 | 5 | 7 | 5 | 5 |
| 3.36% | 13.33% | 2.32% | 827 | 21 | 20 | 1 | 806 | 5 | 7 | 5 | 7 |

Parameter Testing for Opaque Colony Count Results for the modified Circular Hough Transform.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| F-Score | Precision | Recall | Actual | Count | TP | FP | FN | MN\_DIST | ACCUM | MN\_RAD | MX\_RAD |
| 0.93% | 0.47% | 100.00% | 827 | 140374 | 661 | 139713 | 0 | 1 | 1 | 1 | 1 |
| 0.93% | 0.47% | 100.00% | 827 | 140374 | 661 | 139713 | 0 | 1 | 1 | 1 | 3 |
| 0.60% | 0.30% | 100.00% | 827 | 228162 | 690 | 227472 | 0 | 1 | 1 | 1 | 5 |
| 0.46% | 0.23% | 100.00% | 827 | 299066 | 699 | 298367 | 0 | 1 | 1 | 1 | 7 |
| 0.97% | 0.49% | 100.00% | 827 | 129619 | 639 | 128980 | 0 | 1 | 1 | 3 | 3 |
| 0.97% | 0.49% | 100.00% | 827 | 129619 | 639 | 128980 | 0 | 1 | 1 | 3 | 5 |
| 0.57% | 0.28% | 100.00% | 827 | 229898 | 659 | 229239 | 0 | 1 | 1 | 3 | 7 |
| 0.65% | 0.33% | 100.00% | 827 | 144803 | 480 | 144323 | 0 | 1 | 1 | 5 | 5 |
| 0.65% | 0.33% | 100.00% | 827 | 144803 | 480 | 144323 | 0 | 1 | 1 | 5 | 7 |
| 1.99% | 1.01% | 100.00% | 827 | 61897 | 626 | 61271 | 0 | 1 | 2 | 1 | 1 |
| 1.99% | 1.01% | 100.00% | 827 | 61897 | 626 | 61271 | 0 | 1 | 2 | 1 | 3 |
| 0.95% | 0.48% | 100.00% | 827 | 138988 | 669 | 138319 | 0 | 1 | 2 | 1 | 5 |
| 0.63% | 0.32% | 100.00% | 827 | 209772 | 675 | 209097 | 0 | 1 | 2 | 1 | 7 |
| 2.19% | 1.11% | 100.00% | 827 | 52973 | 593 | 52380 | 0 | 1 | 2 | 3 | 3 |
| 2.19% | 1.11% | 100.00% | 827 | 52973 | 593 | 52380 | 0 | 1 | 2 | 3 | 5 |
| 0.91% | 0.46% | 100.00% | 827 | 134332 | 619 | 133713 | 0 | 1 | 2 | 3 | 7 |
| 1.23% | 0.62% | 100.00% | 827 | 65214 | 410 | 64804 | 0 | 1 | 2 | 5 | 5 |
| 1.23% | 0.62% | 100.00% | 827 | 65214 | 410 | 64804 | 0 | 1 | 2 | 5 | 7 |
| 5.09% | 2.63% | 100.00% | 827 | 22918 | 598 | 22320 | 0 | 1 | 3 | 1 | 1 |
| 5.09% | 2.63% | 100.00% | 827 | 22918 | 598 | 22320 | 0 | 1 | 3 | 1 | 3 |
| 1.78% | 0.90% | 100.00% | 827 | 71357 | 649 | 70708 | 0 | 1 | 3 | 1 | 5 |
| 0.99% | 0.50% | 100.00% | 827 | 130473 | 656 | 129817 | 0 | 1 | 3 | 1 | 7 |
| 5.43% | 2.81% | 100.00% | 827 | 19877 | 560 | 19317 | 0 | 1 | 3 | 3 | 3 |
| 5.43% | 2.81% | 100.00% | 827 | 19877 | 560 | 19317 | 0 | 1 | 3 | 3 | 5 |
| 1.59% | 0.80% | 100.00% | 827 | 71739 | 583 | 71156 | 0 | 1 | 3 | 3 | 7 |
| 2.45% | 1.25% | 100.00% | 827 | 26690 | 336 | 26354 | 0 | 1 | 3 | 5 | 5 |
| 2.45% | 1.25% | 100.00% | 827 | 26690 | 336 | 26354 | 0 | 1 | 3 | 5 | 7 |
| 11.94% | 6.43% | 100.00% | 827 | 8537 | 550 | 7987 | 0 | 1 | 4 | 1 | 1 |
| 11.94% | 6.43% | 100.00% | 827 | 8537 | 550 | 7987 | 0 | 1 | 4 | 1 | 3 |
| 3.66% | 1.87% | 100.00% | 827 | 33043 | 621 | 32422 | 0 | 1 | 4 | 1 | 5 |
| 1.65% | 0.83% | 100.00% | 827 | 75512 | 638 | 74874 | 0 | 1 | 4 | 1 | 7 |
| 12.02% | 6.47% | 100.00% | 827 | 7710 | 498 | 7212 | 0 | 1 | 4 | 3 | 3 |
| 12.02% | 6.47% | 100.00% | 827 | 7710 | 498 | 7212 | 0 | 1 | 4 | 3 | 5 |
| 2.86% | 1.45% | 100.00% | 827 | 36214 | 532 | 35682 | 0 | 1 | 4 | 3 | 7 |
| 4.92% | 2.55% | 96.67% | 827 | 10608 | 275 | 10333 | 0 | 1 | 4 | 5 | 5 |
| 4.92% | 2.55% | 96.67% | 827 | 10608 | 275 | 10333 | 0 | 1 | 4 | 5 | 7 |
| 25.22% | 14.86% | 100.00% | 827 | 3236 | 484 | 2752 | 0 | 1 | 5 | 1 | 1 |
| 25.22% | 14.86% | 100.00% | 827 | 3236 | 484 | 2752 | 0 | 1 | 5 | 1 | 3 |
| 7.59% | 3.98% | 100.00% | 827 | 14749 | 586 | 14163 | 0 | 1 | 5 | 1 | 5 |
| 2.88% | 1.47% | 100.00% | 827 | 41447 | 614 | 40833 | 0 | 1 | 5 | 1 | 7 |
| 23.35% | 13.59% | 100.00% | 827 | 3216 | 444 | 2772 | 0 | 1 | 5 | 3 | 3 |
| 23.35% | 13.59% | 100.00% | 827 | 3216 | 444 | 2772 | 0 | 1 | 5 | 3 | 5 |
| 5.34% | 2.76% | 100.00% | 827 | 17685 | 494 | 17191 | 0 | 1 | 5 | 3 | 7 |
| 8.59% | 4.62% | 93.33% | 827 | 4315 | 206 | 4109 | 0 | 1 | 5 | 5 | 5 |
| 8.59% | 4.62% | 93.33% | 827 | 4315 | 206 | 4109 | 0 | 1 | 5 | 5 | 7 |
| 45.22% | 30.87% | 98.15% | 827 | 1307 | 423 | 884 | 15 | 1 | 6 | 1 | 1 |
| 45.22% | 30.87% | 98.15% | 827 | 1307 | 423 | 884 | 15 | 1 | 6 | 1 | 3 |
| 15.33% | 8.44% | 100.00% | 827 | 6635 | 560 | 6075 | 0 | 1 | 6 | 1 | 5 |
| 5.34% | 2.76% | 100.00% | 827 | 21256 | 588 | 20668 | 0 | 1 | 6 | 1 | 7 |
| 40.34% | 26.75% | 99.01% | 827 | 1461 | 406 | 1055 | 14 | 1 | 6 | 3 | 3 |
| 40.34% | 26.75% | 99.01% | 827 | 1461 | 406 | 1055 | 14 | 1 | 6 | 3 | 5 |
| 9.65% | 5.13% | 100.00% | 827 | 8688 | 448 | 8240 | 0 | 1 | 6 | 3 | 7 |
| 13.37% | 7.61% | 88.33% | 827 | 1783 | 150 | 1633 | 7 | 1 | 6 | 5 | 5 |
| 13.37% | 7.61% | 88.33% | 827 | 1783 | 150 | 1633 | 7 | 1 | 6 | 5 | 7 |
| 55.84% | 54.94% | 67.15% | 827 | 590 | 339 | 251 | 253 | 1 | 7 | 1 | 1 |
| 55.84% | 54.94% | 67.15% | 827 | 590 | 339 | 251 | 253 | 1 | 7 | 1 | 3 |
| 28.26% | 16.88% | 100.00% | 827 | 3106 | 525 | 2581 | 0 | 1 | 7 | 1 | 5 |
| 9.60% | 5.09% | 100.00% | 827 | 10887 | 551 | 10336 | 0 | 1 | 7 | 1 | 7 |
| 55.23% | 45.85% | 80.07% | 827 | 746 | 369 | 377 | 155 | 1 | 7 | 3 | 3 |
| 55.23% | 45.85% | 80.07% | 827 | 746 | 369 | 377 | 155 | 1 | 7 | 3 | 5 |
| 15.90% | 8.82% | 100.00% | 827 | 4476 | 399 | 4077 | 0 | 1 | 7 | 3 | 7 |
| 15.57% | 11.08% | 48.24% | 827 | 749 | 100 | 649 | 206 | 1 | 7 | 5 | 5 |
| 15.57% | 11.08% | 48.24% | 827 | 749 | 100 | 649 | 206 | 1 | 7 | 5 | 7 |
| 1.91% | 0.97% | 100.00% | 827 | 64508 | 629 | 63879 | 0 | 3 | 1 | 1 | 1 |
| 1.91% | 0.97% | 100.00% | 827 | 64508 | 629 | 63879 | 0 | 3 | 1 | 1 | 3 |
| 1.40% | 0.71% | 100.00% | 827 | 89713 | 638 | 89075 | 0 | 3 | 1 | 1 | 5 |
| 1.11% | 0.56% | 100.00% | 827 | 111882 | 628 | 111254 | 0 | 3 | 1 | 1 | 7 |
| 1.70% | 0.86% | 100.00% | 827 | 65361 | 564 | 64797 | 0 | 3 | 1 | 3 | 3 |
| 1.70% | 0.86% | 100.00% | 827 | 65361 | 564 | 64797 | 0 | 3 | 1 | 3 | 5 |
| 1.11% | 0.56% | 100.00% | 827 | 93604 | 527 | 93077 | 0 | 3 | 1 | 3 | 7 |
| 0.95% | 0.48% | 100.00% | 827 | 72853 | 349 | 72504 | 0 | 3 | 1 | 5 | 5 |
| 0.95% | 0.48% | 100.00% | 827 | 72853 | 349 | 72504 | 0 | 3 | 1 | 5 | 7 |
| 3.38% | 1.73% | 100.00% | 827 | 34559 | 601 | 33958 | 0 | 3 | 2 | 1 | 1 |
| 3.38% | 1.73% | 100.00% | 827 | 34559 | 601 | 33958 | 0 | 3 | 2 | 1 | 3 |
| 2.03% | 1.03% | 100.00% | 827 | 61187 | 634 | 60553 | 0 | 3 | 2 | 1 | 5 |
| 1.44% | 0.73% | 100.00% | 827 | 84356 | 619 | 83737 | 0 | 3 | 2 | 1 | 7 |
| 3.40% | 1.74% | 100.00% | 827 | 31433 | 548 | 30885 | 0 | 3 | 2 | 3 | 3 |
| 3.40% | 1.74% | 100.00% | 827 | 31433 | 548 | 30885 | 0 | 3 | 2 | 3 | 5 |
| 1.61% | 0.82% | 100.00% | 827 | 62074 | 509 | 61565 | 0 | 3 | 2 | 3 | 7 |
| 1.66% | 0.84% | 100.00% | 827 | 39295 | 330 | 38965 | 0 | 3 | 2 | 5 | 5 |
| 1.66% | 0.84% | 100.00% | 827 | 39295 | 330 | 38965 | 0 | 3 | 2 | 5 | 7 |
| 7.17% | 3.74% | 100.00% | 827 | 15564 | 582 | 14982 | 0 | 3 | 3 | 1 | 1 |
| 7.17% | 3.74% | 100.00% | 827 | 15564 | 582 | 14982 | 0 | 3 | 3 | 1 | 3 |
| 3.35% | 1.71% | 100.00% | 827 | 36130 | 623 | 35507 | 0 | 3 | 3 | 1 | 5 |
| 2.05% | 1.04% | 100.00% | 827 | 57785 | 604 | 57181 | 0 | 3 | 3 | 1 | 7 |
| 7.35% | 3.84% | 100.00% | 827 | 13820 | 534 | 13286 | 0 | 3 | 3 | 3 | 3 |
| 7.35% | 3.84% | 100.00% | 827 | 13820 | 534 | 13286 | 0 | 3 | 3 | 3 | 5 |
| 2.62% | 1.33% | 100.00% | 827 | 37195 | 501 | 36694 | 0 | 3 | 3 | 3 | 7 |
| 3.05% | 1.56% | 100.00% | 827 | 18573 | 291 | 18282 | 0 | 3 | 3 | 5 | 5 |
| 3.05% | 1.56% | 100.00% | 827 | 18573 | 291 | 18282 | 0 | 3 | 3 | 5 | 7 |
| 14.50% | 7.94% | 100.00% | 827 | 6747 | 538 | 6209 | 0 | 3 | 4 | 1 | 1 |
| 14.50% | 7.94% | 100.00% | 827 | 6747 | 538 | 6209 | 0 | 3 | 4 | 1 | 3 |
| 6.02% | 3.12% | 100.00% | 827 | 19213 | 605 | 18608 | 0 | 3 | 4 | 1 | 5 |
| 3.16% | 1.61% | 100.00% | 827 | 36382 | 593 | 35789 | 0 | 3 | 4 | 1 | 7 |
| 15.37% | 8.45% | 100.00% | 827 | 5816 | 491 | 5325 | 0 | 3 | 4 | 3 | 3 |
| 15.37% | 8.45% | 100.00% | 827 | 5816 | 491 | 5325 | 0 | 3 | 4 | 3 | 5 |
| 4.37% | 2.25% | 100.00% | 827 | 21154 | 480 | 20674 | 0 | 3 | 4 | 3 | 7 |
| 5.89% | 3.08% | 96.67% | 827 | 8148 | 252 | 7896 | 0 | 3 | 4 | 5 | 5 |
| 5.89% | 3.08% | 96.67% | 827 | 8148 | 252 | 7896 | 0 | 3 | 4 | 5 | 7 |
| 27.56% | 16.50% | 100.00% | 827 | 2873 | 480 | 2393 | 0 | 3 | 5 | 1 | 1 |
| 27.56% | 16.50% | 100.00% | 827 | 2873 | 480 | 2393 | 0 | 3 | 5 | 1 | 3 |
| 11.23% | 6.02% | 100.00% | 827 | 9543 | 578 | 8965 | 0 | 3 | 5 | 1 | 5 |
| 5.05% | 2.61% | 100.00% | 827 | 22105 | 581 | 21524 | 0 | 3 | 5 | 1 | 7 |
| 27.22% | 16.26% | 100.00% | 827 | 2647 | 442 | 2205 | 0 | 3 | 5 | 3 | 3 |
| 27.22% | 16.26% | 100.00% | 827 | 2647 | 442 | 2205 | 0 | 3 | 5 | 3 | 5 |
| 7.46% | 3.91% | 100.00% | 827 | 11365 | 449 | 10916 | 0 | 3 | 5 | 3 | 7 |
| 10.00% | 5.45% | 93.33% | 827 | 3519 | 197 | 3322 | 0 | 3 | 5 | 5 | 5 |
| 10.00% | 5.45% | 93.33% | 827 | 3519 | 197 | 3322 | 0 | 3 | 5 | 5 | 7 |
| 46.63% | 32.35% | 97.02% | 827 | 1246 | 422 | 824 | 23 | 3 | 6 | 1 | 1 |
| 46.63% | 32.35% | 97.02% | 827 | 1246 | 422 | 824 | 23 | 3 | 6 | 1 | 3 |
| 20.30% | 11.53% | 100.00% | 827 | 4801 | 556 | 4245 | 0 | 3 | 6 | 1 | 5 |
| 8.47% | 4.47% | 100.00% | 827 | 12676 | 568 | 12108 | 0 | 3 | 6 | 1 | 7 |
| 43.31% | 29.60% | 97.63% | 827 | 1316 | 407 | 909 | 25 | 3 | 6 | 3 | 3 |
| 43.31% | 29.60% | 97.63% | 827 | 1316 | 407 | 909 | 25 | 3 | 6 | 3 | 5 |
| 13.16% | 7.17% | 100.00% | 827 | 5883 | 423 | 5460 | 0 | 3 | 6 | 3 | 7 |
| 14.96% | 8.67% | 87.78% | 827 | 1529 | 146 | 1383 | 14 | 3 | 6 | 5 | 5 |
| 14.96% | 8.67% | 87.78% | 827 | 1529 | 146 | 1383 | 14 | 3 | 6 | 5 | 7 |
| 55.95% | 55.63% | 66.16% | 827 | 583 | 339 | 244 | 259 | 3 | 7 | 1 | 1 |
| 55.95% | 55.63% | 66.16% | 827 | 583 | 339 | 244 | 259 | 3 | 7 | 1 | 3 |
| 33.20% | 20.47% | 100.00% | 827 | 2532 | 525 | 2007 | 0 | 3 | 7 | 1 | 5 |
| 14.00% | 7.64% | 100.00% | 827 | 7055 | 539 | 6516 | 0 | 3 | 7 | 1 | 7 |
| 56.51% | 48.50% | 77.44% | 827 | 701 | 369 | 332 | 178 | 3 | 7 | 3 | 3 |
| 56.51% | 48.50% | 77.44% | 827 | 701 | 369 | 332 | 178 | 3 | 7 | 3 | 5 |
| 20.98% | 12.06% | 100.00% | 827 | 3149 | 387 | 2762 | 0 | 3 | 7 | 3 | 7 |
| 16.45% | 12.28% | 45.73% | 827 | 676 | 99 | 577 | 237 | 3 | 7 | 5 | 5 |
| 16.45% | 12.28% | 45.73% | 827 | 676 | 99 | 577 | 237 | 3 | 7 | 5 | 7 |
| 2.95% | 1.50% | 100.00% | 827 | 38179 | 579 | 37600 | 0 | 5 | 1 | 1 | 1 |
| 2.95% | 1.50% | 100.00% | 827 | 38179 | 579 | 37600 | 0 | 5 | 1 | 1 | 3 |
| 2.39% | 1.21% | 100.00% | 827 | 49367 | 602 | 48765 | 0 | 5 | 1 | 1 | 5 |
| 1.86% | 0.94% | 100.00% | 827 | 57272 | 543 | 56729 | 0 | 5 | 1 | 1 | 7 |
| 2.52% | 1.28% | 100.00% | 827 | 40629 | 520 | 40109 | 0 | 5 | 1 | 3 | 3 |
| 2.52% | 1.28% | 100.00% | 827 | 40629 | 520 | 40109 | 0 | 5 | 1 | 3 | 5 |
| 1.63% | 0.83% | 100.00% | 827 | 51729 | 428 | 51301 | 0 | 5 | 1 | 3 | 7 |
| 0.99% | 0.50% | 96.67% | 827 | 45142 | 221 | 44921 | 0 | 5 | 1 | 5 | 5 |
| 0.99% | 0.50% | 96.67% | 827 | 45142 | 221 | 44921 | 0 | 5 | 1 | 5 | 7 |
| 4.82% | 2.48% | 100.00% | 827 | 22762 | 570 | 22192 | 0 | 5 | 2 | 1 | 1 |
| 4.82% | 2.48% | 100.00% | 827 | 22762 | 570 | 22192 | 0 | 5 | 2 | 1 | 3 |
| 3.27% | 1.67% | 100.00% | 827 | 36098 | 608 | 35490 | 0 | 5 | 2 | 1 | 5 |
| 2.37% | 1.20% | 100.00% | 827 | 45534 | 551 | 44983 | 0 | 5 | 2 | 1 | 7 |
| 4.61% | 2.37% | 100.00% | 827 | 22034 | 523 | 21511 | 0 | 5 | 2 | 3 | 3 |
| 4.61% | 2.37% | 100.00% | 827 | 22034 | 523 | 21511 | 0 | 5 | 2 | 3 | 5 |
| 2.33% | 1.18% | 100.00% | 827 | 37670 | 445 | 37225 | 0 | 5 | 2 | 3 | 7 |
| 1.62% | 0.82% | 100.00% | 827 | 27335 | 219 | 27116 | 0 | 5 | 2 | 5 | 5 |
| 1.62% | 0.82% | 100.00% | 827 | 27335 | 219 | 27116 | 0 | 5 | 2 | 5 | 7 |
| 9.19% | 4.86% | 100.00% | 827 | 11405 | 557 | 10848 | 0 | 5 | 3 | 1 | 1 |
| 9.19% | 4.86% | 100.00% | 827 | 11405 | 557 | 10848 | 0 | 5 | 3 | 1 | 3 |
| 5.01% | 2.58% | 100.00% | 827 | 23214 | 607 | 22607 | 0 | 5 | 3 | 1 | 5 |
| 3.25% | 1.66% | 100.00% | 827 | 33300 | 556 | 32744 | 0 | 5 | 3 | 1 | 7 |
| 9.17% | 4.85% | 100.00% | 827 | 10733 | 523 | 10210 | 0 | 5 | 3 | 3 | 3 |
| 9.17% | 4.85% | 100.00% | 827 | 10733 | 523 | 10210 | 0 | 5 | 3 | 3 | 5 |
| 3.59% | 1.84% | 100.00% | 827 | 24311 | 449 | 23862 | 0 | 5 | 3 | 3 | 7 |
| 3.09% | 1.59% | 96.67% | 827 | 14267 | 225 | 14042 | 0 | 5 | 3 | 5 | 5 |
| 3.09% | 1.59% | 96.67% | 827 | 14267 | 225 | 14042 | 0 | 5 | 3 | 5 | 7 |
| 17.36% | 9.67% | 100.00% | 827 | 5403 | 528 | 4875 | 0 | 5 | 4 | 1 | 1 |
| 17.36% | 9.67% | 100.00% | 827 | 5403 | 528 | 4875 | 0 | 5 | 4 | 1 | 3 |
| 8.31% | 4.38% | 100.00% | 827 | 13442 | 597 | 12845 | 0 | 5 | 4 | 1 | 5 |
| 4.81% | 2.48% | 100.00% | 827 | 22168 | 556 | 21612 | 0 | 5 | 4 | 1 | 7 |
| 17.88% | 10.00% | 100.00% | 827 | 4853 | 485 | 4368 | 0 | 5 | 4 | 3 | 3 |
| 17.88% | 10.00% | 100.00% | 827 | 4853 | 485 | 4368 | 0 | 5 | 4 | 3 | 5 |
| 5.75% | 2.98% | 100.00% | 827 | 14893 | 448 | 14445 | 0 | 5 | 4 | 3 | 7 |
| 6.06% | 3.19% | 96.67% | 827 | 6707 | 212 | 6495 | 0 | 5 | 4 | 5 | 5 |
| 6.06% | 3.19% | 96.67% | 827 | 6707 | 212 | 6495 | 0 | 5 | 4 | 5 | 7 |
| 29.96% | 18.18% | 100.00% | 827 | 2557 | 477 | 2080 | 0 | 5 | 5 | 1 | 1 |
| 29.96% | 18.18% | 100.00% | 827 | 2557 | 477 | 2080 | 0 | 5 | 5 | 1 | 3 |
| 14.24% | 7.78% | 100.00% | 827 | 7250 | 573 | 6677 | 0 | 5 | 5 | 1 | 5 |
| 7.34% | 3.84% | 100.00% | 827 | 14306 | 557 | 13749 | 0 | 5 | 5 | 1 | 7 |
| 29.53% | 17.91% | 100.00% | 827 | 2386 | 440 | 1946 | 0 | 5 | 5 | 3 | 3 |
| 29.53% | 17.91% | 100.00% | 827 | 2386 | 440 | 1946 | 0 | 5 | 5 | 3 | 5 |
| 9.33% | 4.95% | 100.00% | 827 | 8601 | 432 | 8169 | 0 | 5 | 5 | 3 | 7 |
| 10.84% | 5.96% | 93.33% | 827 | 3060 | 184 | 2876 | 0 | 5 | 5 | 5 | 5 |
| 10.84% | 5.96% | 93.33% | 827 | 3060 | 184 | 2876 | 0 | 5 | 5 | 5 | 7 |
| 47.53% | 33.17% | 96.80% | 827 | 1206 | 422 | 784 | 25 | 5 | 6 | 1 | 1 |
| 47.53% | 33.17% | 96.80% | 827 | 1206 | 422 | 784 | 25 | 5 | 6 | 1 | 3 |
| 24.01% | 13.96% | 100.00% | 827 | 3900 | 553 | 3347 | 0 | 5 | 6 | 1 | 5 |
| 11.39% | 6.12% | 100.00% | 827 | 8936 | 554 | 8382 | 0 | 5 | 6 | 1 | 7 |
| 44.34% | 30.58% | 96.86% | 827 | 1257 | 405 | 852 | 32 | 5 | 6 | 3 | 3 |
| 44.34% | 30.58% | 96.86% | 827 | 1257 | 405 | 852 | 32 | 5 | 6 | 3 | 5 |
| 15.79% | 8.74% | 100.00% | 827 | 4727 | 418 | 4309 | 0 | 5 | 6 | 3 | 7 |
| 15.16% | 8.88% | 83.65% | 827 | 1415 | 138 | 1277 | 23 | 5 | 6 | 5 | 5 |
| 15.16% | 8.88% | 83.65% | 827 | 1415 | 138 | 1277 | 23 | 5 | 6 | 5 | 7 |
| 55.99% | 56.06% | 65.79% | 827 | 578 | 339 | 239 | 263 | 5 | 7 | 1 | 1 |
| 55.99% | 56.06% | 65.79% | 827 | 578 | 339 | 239 | 263 | 5 | 7 | 1 | 3 |
| 36.33% | 22.85% | 100.00% | 827 | 2245 | 524 | 1721 | 0 | 5 | 7 | 1 | 5 |
| 17.70% | 9.88% | 100.00% | 827 | 5299 | 530 | 4769 | 0 | 5 | 7 | 1 | 7 |
| 56.61% | 48.79% | 76.76% | 827 | 693 | 368 | 325 | 183 | 5 | 7 | 3 | 3 |
| 56.61% | 48.79% | 76.76% | 827 | 693 | 368 | 325 | 183 | 5 | 7 | 3 | 5 |
| 24.00% | 14.08% | 100.00% | 827 | 2647 | 381 | 2266 | 0 | 5 | 7 | 3 | 7 |
| 16.57% | 12.63% | 44.31% | 827 | 650 | 98 | 552 | 254 | 5 | 7 | 5 | 5 |
| 16.57% | 12.63% | 44.31% | 827 | 650 | 98 | 552 | 254 | 5 | 7 | 5 | 7 |

Parameter Testing for Translucent Colony Count Results for the modified Circular Hough Transform.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| F-Score | Precision | Recall | Actual | Count | TP | FP | FN | MN\_DIST | ACCUM | MN\_RAD | MX\_RAD |
| 46.05% | 38.81% | 66.54% | 556 | 499 | 216 | 283 | 97 | 20 | 7 | 13 | 13 |
| 46.05% | 38.81% | 66.54% | 556 | 499 | 216 | 283 | 97 | 20 | 7 | 13 | 15 |
| 27.64% | 16.78% | 96.67% | 556 | 1682 | 292 | 1390 | 0 | 20 | 7 | 13 | 17 |
| 19.38% | 11.12% | 96.67% | 556 | 2824 | 323 | 2501 | 0 | 20 | 7 | 13 | 19 |
| 16.13% | 9.01% | 96.67% | 556 | 3638 | 335 | 3303 | 0 | 20 | 7 | 13 | 21 |
| 14.27% | 7.89% | 96.67% | 556 | 4220 | 337 | 3883 | 0 | 20 | 7 | 13 | 23 |
| 46.59% | 41.22% | 59.90% | 556 | 461 | 216 | 245 | 116 | 20 | 7 | 15 | 15 |
| 46.59% | 41.22% | 59.90% | 556 | 461 | 216 | 245 | 116 | 20 | 7 | 15 | 17 |
| 25.31% | 15.27% | 96.67% | 556 | 1708 | 274 | 1434 | 0 | 20 | 7 | 15 | 19 |
| 19.13% | 10.94% | 96.67% | 556 | 2688 | 303 | 2385 | 0 | 20 | 7 | 15 | 21 |
| 15.65% | 8.73% | 96.67% | 556 | 3519 | 313 | 3206 | 0 | 20 | 7 | 15 | 23 |
| 43.66% | 36.29% | 65.17% | 556 | 504 | 213 | 291 | 92 | 20 | 7 | 17 | 17 |
| 43.66% | 36.29% | 65.17% | 556 | 504 | 213 | 291 | 92 | 20 | 7 | 17 | 19 |
| 25.61% | 15.50% | 96.67% | 556 | 1615 | 262 | 1353 | 0 | 20 | 7 | 17 | 21 |
| 18.50% | 10.55% | 96.67% | 556 | 2648 | 287 | 2361 | 0 | 20 | 7 | 17 | 23 |
| 38.92% | 34.57% | 55.31% | 556 | 446 | 181 | 265 | 128 | 20 | 7 | 19 | 19 |
| 38.92% | 34.57% | 55.31% | 556 | 446 | 181 | 265 | 128 | 20 | 7 | 19 | 21 |
| 24.32% | 14.60% | 96.67% | 556 | 1551 | 240 | 1311 | 0 | 20 | 7 | 19 | 23 |
| 33.81% | 30.96% | 43.60% | 556 | 400 | 150 | 250 | 162 | 20 | 7 | 21 | 21 |
| 33.81% | 30.96% | 43.60% | 556 | 400 | 150 | 250 | 162 | 20 | 7 | 21 | 23 |
| 39.30% | 55.51% | 32.86% | 556 | 264 | 170 | 94 | 292 | 20 | 8 | 13 | 13 |
| 39.30% | 55.51% | 32.86% | 556 | 264 | 170 | 94 | 292 | 20 | 8 | 13 | 15 |
| 37.18% | 24.64% | 96.67% | 556 | 968 | 251 | 717 | 0 | 20 | 8 | 13 | 17 |
| 23.76% | 14.13% | 96.67% | 556 | 1980 | 287 | 1693 | 0 | 20 | 8 | 13 | 19 |
| 19.22% | 11.01% | 96.67% | 556 | 2803 | 315 | 2488 | 0 | 20 | 8 | 13 | 21 |
| 16.32% | 9.14% | 96.67% | 556 | 3409 | 318 | 3091 | 0 | 20 | 8 | 13 | 23 |
| 38.39% | 53.38% | 31.84% | 556 | 257 | 171 | 86 | 300 | 20 | 8 | 15 | 15 |
| 38.39% | 53.38% | 31.84% | 556 | 257 | 171 | 86 | 300 | 20 | 8 | 15 | 17 |
| 33.93% | 22.42% | 92.33% | 556 | 1013 | 242 | 771 | 6 | 20 | 8 | 15 | 19 |
| 23.82% | 14.19% | 96.67% | 556 | 1862 | 273 | 1589 | 0 | 20 | 8 | 15 | 21 |
| 18.25% | 10.39% | 96.67% | 556 | 2693 | 289 | 2404 | 0 | 20 | 8 | 15 | 23 |
| 34.27% | 48.12% | 28.60% | 556 | 255 | 157 | 98 | 304 | 20 | 8 | 17 | 17 |
| 34.27% | 48.12% | 28.60% | 556 | 255 | 157 | 98 | 304 | 20 | 8 | 17 | 19 |
| 33.02% | 21.58% | 95.69% | 556 | 943 | 219 | 724 | 5 | 20 | 8 | 17 | 21 |
| 22.47% | 13.27% | 96.67% | 556 | 1817 | 251 | 1566 | 0 | 20 | 8 | 17 | 23 |
| 31.98% | 52.49% | 24.45% | 556 | 216 | 143 | 73 | 340 | 20 | 8 | 19 | 19 |
| 31.98% | 52.49% | 24.45% | 556 | 216 | 143 | 73 | 340 | 20 | 8 | 19 | 21 |
| 33.38% | 21.89% | 92.67% | 556 | 843 | 204 | 639 | 3 | 20 | 8 | 19 | 23 |
| 22.83% | 38.72% | 18.06% | 556 | 195 | 100 | 95 | 362 | 20 | 8 | 21 | 21 |
| 22.83% | 38.72% | 18.06% | 556 | 195 | 100 | 95 | 362 | 20 | 8 | 21 | 23 |
| 32.37% | 77.15% | 22.38% | 556 | 162 | 134 | 28 | 394 | 20 | 9 | 13 | 13 |
| 32.37% | 77.15% | 22.38% | 556 | 162 | 134 | 28 | 394 | 20 | 9 | 13 | 15 |
| 41.92% | 33.79% | 73.60% | 556 | 540 | 200 | 340 | 78 | 20 | 9 | 13 | 17 |
| 29.04% | 18.16% | 96.67% | 556 | 1341 | 249 | 1092 | 0 | 20 | 9 | 13 | 19 |
| 21.66% | 12.73% | 96.67% | 556 | 2079 | 272 | 1807 | 0 | 20 | 9 | 13 | 21 |
| 18.70% | 10.71% | 96.67% | 556 | 2678 | 294 | 2384 | 0 | 20 | 9 | 13 | 23 |
| 31.17% | 66.23% | 21.79% | 556 | 163 | 135 | 28 | 393 | 20 | 9 | 15 | 15 |
| 31.17% | 66.23% | 21.79% | 556 | 163 | 135 | 28 | 393 | 20 | 9 | 15 | 17 |
| 41.19% | 31.62% | 75.43% | 556 | 563 | 202 | 361 | 54 | 20 | 9 | 15 | 19 |
| 30.38% | 19.05% | 96.67% | 556 | 1229 | 244 | 985 | 0 | 20 | 9 | 15 | 21 |
| 22.50% | 13.28% | 96.67% | 556 | 1955 | 268 | 1687 | 0 | 20 | 9 | 15 | 23 |
| 27.11% | 64.92% | 18.91% | 556 | 148 | 120 | 28 | 408 | 20 | 9 | 17 | 17 |
| 27.11% | 64.92% | 18.91% | 556 | 148 | 120 | 28 | 408 | 20 | 9 | 17 | 19 |
| 39.67% | 32.44% | 68.06% | 556 | 494 | 183 | 311 | 95 | 20 | 9 | 17 | 21 |
| 29.01% | 18.12% | 96.67% | 556 | 1153 | 221 | 932 | 0 | 20 | 9 | 17 | 23 |
| 23.86% | 60.44% | 15.92% | 556 | 130 | 104 | 26 | 426 | 20 | 9 | 19 | 19 |
| 23.86% | 60.44% | 15.92% | 556 | 130 | 104 | 26 | 426 | 20 | 9 | 19 | 21 |
| 34.37% | 29.64% | 49.67% | 556 | 462 | 161 | 301 | 123 | 20 | 9 | 19 | 23 |
| 17.35% | 50.91% | 11.14% | 556 | 100 | 73 | 27 | 456 | 20 | 9 | 21 | 21 |
| 17.35% | 50.91% | 11.14% | 556 | 100 | 73 | 27 | 456 | 20 | 9 | 21 | 23 |
| 27.28% | 80.92% | 17.50% | 556 | 117 | 109 | 8 | 439 | 20 | 10 | 13 | 13 |
| 27.28% | 80.92% | 17.50% | 556 | 117 | 109 | 8 | 439 | 20 | 10 | 13 | 15 |
| 37.53% | 46.17% | 33.93% | 556 | 292 | 163 | 129 | 264 | 20 | 10 | 13 | 17 |
| 36.43% | 24.66% | 94.13% | 556 | 809 | 215 | 594 | 7 | 20 | 10 | 13 | 19 |
| 26.11% | 16.02% | 96.67% | 556 | 1460 | 242 | 1218 | 0 | 20 | 10 | 13 | 21 |
| 22.32% | 13.15% | 96.67% | 556 | 1985 | 268 | 1717 | 0 | 20 | 10 | 13 | 23 |
| 24.23% | 68.89% | 15.69% | 556 | 108 | 103 | 5 | 448 | 20 | 10 | 15 | 15 |
| 24.23% | 68.89% | 15.69% | 556 | 108 | 103 | 5 | 448 | 20 | 10 | 15 | 17 |
| 35.85% | 39.85% | 37.67% | 556 | 320 | 161 | 159 | 239 | 20 | 10 | 15 | 19 |
| 34.65% | 23.87% | 88.33% | 556 | 768 | 199 | 569 | 12 | 20 | 10 | 15 | 21 |
| 27.65% | 17.09% | 96.67% | 556 | 1328 | 236 | 1092 | 0 | 20 | 10 | 15 | 23 |
| 22.54% | 69.19% | 14.75% | 556 | 106 | 98 | 8 | 450 | 20 | 10 | 17 | 17 |
| 22.54% | 69.19% | 14.75% | 556 | 106 | 98 | 8 | 450 | 20 | 10 | 17 | 19 |
| 34.16% | 44.66% | 30.05% | 556 | 280 | 152 | 128 | 276 | 20 | 10 | 17 | 21 |
| 35.02% | 24.14% | 87.84% | 556 | 686 | 186 | 500 | 19 | 20 | 10 | 17 | 23 |
| 18.14% | 57.96% | 11.51% | 556 | 84 | 80 | 4 | 472 | 20 | 10 | 19 | 19 |
| 18.14% | 57.96% | 11.51% | 556 | 84 | 80 | 4 | 472 | 20 | 10 | 19 | 21 |
| 27.41% | 39.15% | 22.70% | 556 | 241 | 121 | 120 | 315 | 20 | 10 | 19 | 23 |
| 13.80% | 58.22% | 8.23% | 556 | 60 | 54 | 6 | 496 | 20 | 10 | 21 | 21 |
| 13.80% | 58.22% | 8.23% | 556 | 60 | 54 | 6 | 496 | 20 | 10 | 21 | 23 |
| 23.68% | 81.67% | 14.69% | 556 | 93 | 91 | 2 | 463 | 20 | 11 | 13 | 13 |
| 23.68% | 81.67% | 14.69% | 556 | 93 | 91 | 2 | 463 | 20 | 11 | 13 | 15 |
| 32.67% | 62.88% | 23.30% | 556 | 179 | 138 | 41 | 377 | 20 | 11 | 13 | 17 |
| 38.70% | 33.08% | 64.75% | 556 | 490 | 181 | 309 | 107 | 20 | 11 | 13 | 19 |
| 31.68% | 20.49% | 96.47% | 556 | 979 | 214 | 765 | 1 | 20 | 11 | 13 | 21 |
| 26.02% | 15.83% | 96.67% | 556 | 1446 | 238 | 1208 | 0 | 20 | 11 | 13 | 23 |
| 19.98% | 66.67% | 12.57% | 556 | 84 | 83 | 1 | 472 | 20 | 11 | 15 | 15 |
| 19.98% | 66.67% | 12.57% | 556 | 84 | 83 | 1 | 472 | 20 | 11 | 15 | 17 |
| 29.43% | 48.63% | 22.40% | 556 | 192 | 131 | 61 | 365 | 20 | 11 | 15 | 19 |
| 35.65% | 30.19% | 58.04% | 556 | 467 | 166 | 301 | 116 | 20 | 11 | 15 | 21 |
| 31.53% | 20.77% | 92.28% | 556 | 898 | 200 | 698 | 6 | 20 | 11 | 15 | 23 |
| 17.05% | 61.67% | 10.68% | 556 | 75 | 73 | 2 | 481 | 20 | 11 | 17 | 17 |
| 17.05% | 61.67% | 10.68% | 556 | 75 | 73 | 2 | 481 | 20 | 11 | 17 | 19 |
| 27.34% | 54.40% | 19.92% | 556 | 174 | 120 | 54 | 382 | 20 | 11 | 17 | 21 |
| 34.00% | 32.15% | 43.08% | 556 | 400 | 152 | 248 | 174 | 20 | 11 | 17 | 23 |
| 12.72% | 42.92% | 7.89% | 556 | 57 | 56 | 1 | 499 | 20 | 11 | 19 | 19 |
| 12.72% | 42.92% | 7.89% | 556 | 57 | 56 | 1 | 499 | 20 | 11 | 19 | 21 |
| 22.91% | 52.02% | 15.81% | 556 | 143 | 99 | 44 | 413 | 20 | 11 | 19 | 23 |
| 9.48% | 48.89% | 5.42% | 556 | 39 | 36 | 3 | 517 | 20 | 11 | 21 | 21 |
| 9.48% | 48.89% | 5.42% | 556 | 39 | 36 | 3 | 517 | 20 | 11 | 21 | 23 |

Parameter Testing Opaque Colony Count Results for Blob Detection.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| F | P | R | ACTUAL | COUNT | TP | FP | FN | MN\_REP | MN\_DIST | MN\_IN | MN\_CNVX |
| 80.08% | 82.57% | 81.52% | 827 | 678 | 564 | 114 | 172 | 2 | 2 | 0.4 | 0.7 |
| 80.70% | 86.70% | 78.89% | 827 | 648 | 563 | 85 | 192 | 2 | 2 | 0.4 | 0.8 |
| 81.58% | 94.62% | 73.37% | 827 | 589 | 559 | 30 | 238 | 2 | 2 | 0.4 | 0.9 |
| 80.99% | 88.15% | 77.79% | 827 | 631 | 560 | 71 | 203 | 2 | 2 | 0.5 | 0.7 |
| 81.17% | 90.54% | 76.11% | 827 | 615 | 558 | 57 | 215 | 2 | 2 | 0.5 | 0.8 |
| 81.21% | 96.24% | 71.77% | 827 | 576 | 555 | 21 | 251 | 2 | 2 | 0.5 | 0.9 |
| 80.35% | 93.37% | 72.61% | 827 | 579 | 545 | 34 | 249 | 2 | 2 | 0.6 | 0.7 |
| 80.20% | 94.86% | 70.92% | 827 | 571 | 544 | 27 | 256 | 2 | 2 | 0.6 | 0.8 |
| 79.97% | 97.08% | 69.18% | 827 | 556 | 541 | 15 | 271 | 2 | 2 | 0.6 | 0.9 |
| 78.34% | 96.73% | 67.28% | 827 | 539 | 522 | 17 | 288 | 2 | 2 | 0.7 | 0.7 |
| 78.34% | 97.47% | 66.62% | 827 | 535 | 522 | 13 | 292 | 2 | 2 | 0.7 | 0.8 |
| 78.09% | 98.41% | 65.90% | 827 | 527 | 519 | 8 | 300 | 2 | 2 | 0.7 | 0.9 |
| 80.08% | 82.57% | 81.52% | 827 | 678 | 564 | 114 | 172 | 2 | 3 | 0.4 | 0.7 |
| 80.70% | 86.70% | 78.89% | 827 | 648 | 563 | 85 | 192 | 2 | 3 | 0.4 | 0.8 |
| 81.58% | 94.62% | 73.37% | 827 | 589 | 559 | 30 | 238 | 2 | 3 | 0.4 | 0.9 |
| 80.99% | 88.15% | 77.79% | 827 | 631 | 560 | 71 | 203 | 2 | 3 | 0.5 | 0.7 |
| 81.17% | 90.54% | 76.11% | 827 | 615 | 558 | 57 | 215 | 2 | 3 | 0.5 | 0.8 |
| 81.21% | 96.24% | 71.77% | 827 | 576 | 555 | 21 | 251 | 2 | 3 | 0.5 | 0.9 |
| 80.35% | 93.37% | 72.61% | 827 | 579 | 545 | 34 | 249 | 2 | 3 | 0.6 | 0.7 |
| 80.20% | 94.86% | 70.92% | 827 | 571 | 544 | 27 | 256 | 2 | 3 | 0.6 | 0.8 |
| 79.97% | 97.08% | 69.18% | 827 | 556 | 541 | 15 | 271 | 2 | 3 | 0.6 | 0.9 |
| 78.34% | 96.73% | 67.28% | 827 | 539 | 522 | 17 | 288 | 2 | 3 | 0.7 | 0.7 |
| 78.34% | 97.47% | 66.62% | 827 | 535 | 522 | 13 | 292 | 2 | 3 | 0.7 | 0.8 |
| 78.09% | 98.41% | 65.90% | 827 | 527 | 519 | 8 | 300 | 2 | 3 | 0.7 | 0.9 |
| 80.49% | 86.83% | 78.22% | 827 | 632 | 552 | 80 | 203 | 3 | 2 | 0.4 | 0.7 |
| 80.61% | 89.79% | 75.83% | 827 | 613 | 551 | 62 | 217 | 3 | 2 | 0.4 | 0.8 |
| 80.53% | 95.32% | 71.12% | 827 | 572 | 547 | 25 | 255 | 3 | 2 | 0.4 | 0.9 |
| 80.42% | 90.51% | 74.80% | 827 | 602 | 548 | 54 | 228 | 3 | 2 | 0.5 | 0.7 |
| 80.37% | 92.45% | 73.16% | 827 | 589 | 546 | 43 | 238 | 3 | 2 | 0.5 | 0.8 |
| 80.16% | 96.66% | 69.77% | 827 | 561 | 543 | 18 | 266 | 3 | 2 | 0.5 | 0.9 |
| 79.27% | 96.09% | 69.03% | 827 | 556 | 533 | 23 | 271 | 3 | 2 | 0.6 | 0.7 |
| 79.05% | 96.50% | 68.26% | 827 | 551 | 532 | 19 | 276 | 3 | 2 | 0.6 | 0.8 |
| 78.87% | 97.82% | 67.23% | 827 | 541 | 530 | 11 | 286 | 3 | 2 | 0.6 | 0.9 |
| 77.54% | 97.78% | 65.69% | 827 | 524 | 511 | 13 | 303 | 3 | 2 | 0.7 | 0.7 |
| 77.54% | 98.16% | 65.32% | 827 | 521 | 511 | 10 | 306 | 3 | 2 | 0.7 | 0.8 |
| 77.23% | 99.03% | 64.60% | 827 | 513 | 507 | 6 | 314 | 3 | 2 | 0.7 | 0.9 |
| 80.49% | 86.83% | 78.22% | 827 | 632 | 552 | 80 | 203 | 3 | 3 | 0.4 | 0.7 |
| 80.61% | 89.79% | 75.83% | 827 | 613 | 551 | 62 | 217 | 3 | 3 | 0.4 | 0.8 |
| 80.53% | 95.32% | 71.12% | 827 | 572 | 547 | 25 | 255 | 3 | 3 | 0.4 | 0.9 |
| 80.42% | 90.51% | 74.80% | 827 | 602 | 548 | 54 | 228 | 3 | 3 | 0.5 | 0.7 |
| 80.37% | 92.45% | 73.16% | 827 | 589 | 546 | 43 | 238 | 3 | 3 | 0.5 | 0.8 |
| 80.16% | 96.66% | 69.77% | 827 | 561 | 543 | 18 | 266 | 3 | 3 | 0.5 | 0.9 |
| 79.27% | 96.09% | 69.03% | 827 | 556 | 533 | 23 | 271 | 3 | 3 | 0.6 | 0.7 |
| 79.05% | 96.50% | 68.26% | 827 | 551 | 532 | 19 | 276 | 3 | 3 | 0.6 | 0.8 |
| 78.87% | 97.82% | 67.23% | 827 | 541 | 530 | 11 | 286 | 3 | 3 | 0.6 | 0.9 |
| 77.54% | 97.78% | 65.69% | 827 | 524 | 511 | 13 | 303 | 3 | 3 | 0.7 | 0.7 |
| 77.54% | 98.16% | 65.32% | 827 | 521 | 511 | 10 | 306 | 3 | 3 | 0.7 | 0.8 |
| 77.23% | 99.03% | 64.60% | 827 | 513 | 507 | 6 | 314 | 3 | 3 | 0.7 | 0.9 |
| 79.30% | 89.81% | 73.35% | 827 | 587 | 532 | 55 | 240 | 4 | 2 | 0.4 | 0.7 |
| 79.27% | 92.16% | 71.46% | 827 | 575 | 531 | 44 | 252 | 4 | 2 | 0.4 | 0.8 |
| 79.17% | 96.81% | 68.14% | 827 | 547 | 530 | 17 | 280 | 4 | 2 | 0.4 | 0.9 |
| 78.97% | 93.46% | 70.68% | 827 | 563 | 527 | 36 | 264 | 4 | 2 | 0.5 | 0.7 |
| 78.97% | 94.46% | 69.64% | 827 | 557 | 527 | 30 | 270 | 4 | 2 | 0.5 | 0.8 |
| 78.74% | 97.78% | 67.08% | 827 | 536 | 524 | 12 | 291 | 4 | 2 | 0.5 | 0.9 |
| 78.04% | 97.42% | 66.25% | 827 | 533 | 518 | 15 | 294 | 4 | 2 | 0.6 | 0.7 |
| 78.04% | 97.70% | 66.09% | 827 | 531 | 518 | 13 | 296 | 4 | 2 | 0.6 | 0.8 |
| 77.88% | 98.72% | 65.31% | 827 | 523 | 516 | 7 | 304 | 4 | 2 | 0.6 | 0.9 |
| 76.63% | 98.86% | 63.71% | 827 | 506 | 500 | 6 | 321 | 4 | 2 | 0.7 | 0.7 |
| 76.63% | 99.16% | 63.55% | 827 | 504 | 500 | 4 | 323 | 4 | 2 | 0.7 | 0.8 |
| 76.46% | 99.41% | 63.29% | 827 | 501 | 498 | 3 | 326 | 4 | 2 | 0.7 | 0.9 |
| 79.30% | 89.81% | 73.35% | 827 | 587 | 532 | 55 | 240 | 4 | 3 | 0.4 | 0.7 |
| 79.27% | 92.16% | 71.46% | 827 | 575 | 531 | 44 | 252 | 4 | 3 | 0.4 | 0.8 |
| 79.17% | 96.81% | 68.14% | 827 | 547 | 530 | 17 | 280 | 4 | 3 | 0.4 | 0.9 |
| 78.97% | 93.46% | 70.68% | 827 | 563 | 527 | 36 | 264 | 4 | 3 | 0.5 | 0.7 |
| 78.97% | 94.46% | 69.64% | 827 | 557 | 527 | 30 | 270 | 4 | 3 | 0.5 | 0.8 |
| 78.74% | 97.78% | 67.08% | 827 | 536 | 524 | 12 | 291 | 4 | 3 | 0.5 | 0.9 |
| 78.04% | 97.42% | 66.25% | 827 | 533 | 518 | 15 | 294 | 4 | 3 | 0.6 | 0.7 |
| 78.04% | 97.70% | 66.09% | 827 | 531 | 518 | 13 | 296 | 4 | 3 | 0.6 | 0.8 |
| 77.88% | 98.72% | 65.31% | 827 | 523 | 516 | 7 | 304 | 4 | 3 | 0.6 | 0.9 |
| 76.63% | 98.86% | 63.71% | 827 | 506 | 500 | 6 | 321 | 4 | 3 | 0.7 | 0.7 |
| 76.63% | 99.16% | 63.55% | 827 | 504 | 500 | 4 | 323 | 4 | 3 | 0.7 | 0.8 |
| 76.46% | 99.41% | 63.29% | 827 | 501 | 498 | 3 | 326 | 4 | 3 | 0.7 | 0.9 |

Parameter Testing for Translucent Colony Count Results for Blob Detection.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| F | P | R | ACTUAL | COUNT | TP | FP | FN | MN\_REP | MN\_DIST | MN\_IN | MN\_CNVX |
| 64.00% | 53.94% | 90.87% | 556 | 565 | 317 | 248 | 53 | 2 | 2 | 0.4 | 0.7 |
| 66.71% | 61.69% | 82.18% | 556 | 490 | 315 | 175 | 92 | 2 | 2 | 0.4 | 0.8 |
| 69.18% | 81.63% | 61.53% | 556 | 364 | 302 | 62 | 192 | 2 | 2 | 0.4 | 0.9 |
| 65.30% | 58.47% | 85.09% | 556 | 500 | 308 | 192 | 84 | 2 | 2 | 0.5 | 0.7 |
| 66.51% | 65.99% | 74.74% | 556 | 442 | 307 | 135 | 127 | 2 | 2 | 0.5 | 0.8 |
| 67.81% | 83.85% | 58.45% | 556 | 341 | 292 | 49 | 215 | 2 | 2 | 0.5 | 0.9 |
| 63.81% | 65.25% | 71.03% | 556 | 415 | 284 | 131 | 151 | 2 | 2 | 0.6 | 0.7 |
| 64.05% | 71.92% | 65.11% | 556 | 377 | 283 | 94 | 185 | 2 | 2 | 0.6 | 0.8 |
| 64.88% | 88.30% | 53.00% | 556 | 303 | 271 | 32 | 253 | 2 | 2 | 0.6 | 0.9 |
| 57.79% | 69.57% | 53.55% | 556 | 335 | 247 | 88 | 225 | 2 | 2 | 0.7 | 0.7 |
| 57.69% | 74.67% | 48.99% | 556 | 312 | 245 | 67 | 247 | 2 | 2 | 0.7 | 0.8 |
| 57.32% | 87.48% | 43.93% | 556 | 268 | 243 | 25 | 288 | 2 | 2 | 0.7 | 0.9 |
| 64.00% | 53.94% | 90.87% | 556 | 565 | 317 | 248 | 53 | 2 | 3 | 0.4 | 0.7 |
| 66.71% | 61.69% | 82.18% | 556 | 490 | 315 | 175 | 92 | 2 | 3 | 0.4 | 0.8 |
| 69.18% | 81.63% | 61.53% | 556 | 364 | 302 | 62 | 192 | 2 | 3 | 0.4 | 0.9 |
| 65.30% | 58.47% | 85.09% | 556 | 500 | 308 | 192 | 84 | 2 | 3 | 0.5 | 0.7 |
| 66.51% | 65.99% | 74.74% | 556 | 442 | 307 | 135 | 127 | 2 | 3 | 0.5 | 0.8 |
| 67.81% | 83.85% | 58.45% | 556 | 341 | 292 | 49 | 215 | 2 | 3 | 0.5 | 0.9 |
| 63.81% | 65.25% | 71.03% | 556 | 415 | 284 | 131 | 151 | 2 | 3 | 0.6 | 0.7 |
| 64.05% | 71.92% | 65.11% | 556 | 377 | 283 | 94 | 185 | 2 | 3 | 0.6 | 0.8 |
| 64.88% | 88.30% | 53.00% | 556 | 303 | 271 | 32 | 253 | 2 | 3 | 0.6 | 0.9 |
| 57.79% | 69.57% | 53.55% | 556 | 335 | 247 | 88 | 225 | 2 | 3 | 0.7 | 0.7 |
| 57.69% | 74.67% | 48.99% | 556 | 312 | 245 | 67 | 247 | 2 | 3 | 0.7 | 0.8 |
| 57.32% | 87.48% | 43.93% | 556 | 268 | 243 | 25 | 288 | 2 | 3 | 0.7 | 0.9 |
| 64.70% | 57.70% | 84.94% | 556 | 509 | 308 | 201 | 81 | 3 | 2 | 0.4 | 0.7 |
| 66.76% | 65.61% | 76.90% | 556 | 446 | 307 | 139 | 120 | 3 | 2 | 0.4 | 0.8 |
| 67.79% | 85.52% | 58.07% | 556 | 341 | 293 | 48 | 215 | 3 | 2 | 0.4 | 0.9 |
| 65.57% | 63.16% | 77.33% | 556 | 451 | 300 | 151 | 120 | 3 | 2 | 0.5 | 0.7 |
| 66.15% | 69.63% | 69.61% | 556 | 406 | 298 | 108 | 156 | 3 | 2 | 0.5 | 0.8 |
| 66.57% | 87.69% | 55.45% | 556 | 320 | 283 | 37 | 236 | 3 | 2 | 0.5 | 0.9 |
| 62.99% | 69.38% | 65.28% | 556 | 380 | 276 | 104 | 183 | 3 | 2 | 0.6 | 0.7 |
| 63.07% | 74.61% | 61.25% | 556 | 353 | 275 | 78 | 208 | 3 | 2 | 0.6 | 0.8 |
| 63.94% | 91.49% | 50.94% | 556 | 288 | 264 | 24 | 268 | 3 | 2 | 0.6 | 0.9 |
| 56.16% | 72.64% | 49.57% | 556 | 306 | 237 | 69 | 251 | 3 | 2 | 0.7 | 0.7 |
| 55.89% | 76.78% | 45.71% | 556 | 289 | 235 | 54 | 268 | 3 | 2 | 0.7 | 0.8 |
| 55.64% | 87.85% | 41.85% | 556 | 256 | 234 | 22 | 300 | 3 | 2 | 0.7 | 0.9 |
| 64.70% | 57.70% | 84.94% | 556 | 509 | 308 | 201 | 81 | 3 | 3 | 0.4 | 0.7 |
| 66.76% | 65.61% | 76.90% | 556 | 446 | 307 | 139 | 120 | 3 | 3 | 0.4 | 0.8 |
| 67.79% | 85.52% | 58.07% | 556 | 341 | 293 | 48 | 215 | 3 | 3 | 0.4 | 0.9 |
| 65.57% | 63.16% | 77.33% | 556 | 451 | 300 | 151 | 120 | 3 | 3 | 0.5 | 0.7 |
| 66.15% | 69.63% | 69.61% | 556 | 406 | 298 | 108 | 156 | 3 | 3 | 0.5 | 0.8 |
| 66.57% | 87.69% | 55.45% | 556 | 320 | 283 | 37 | 236 | 3 | 3 | 0.5 | 0.9 |
| 62.99% | 69.38% | 65.28% | 556 | 380 | 276 | 104 | 183 | 3 | 3 | 0.6 | 0.7 |
| 63.07% | 74.61% | 61.25% | 556 | 353 | 275 | 78 | 208 | 3 | 3 | 0.6 | 0.8 |
| 63.94% | 91.49% | 50.94% | 556 | 288 | 264 | 24 | 268 | 3 | 3 | 0.6 | 0.9 |
| 56.16% | 72.64% | 49.57% | 556 | 306 | 237 | 69 | 251 | 3 | 3 | 0.7 | 0.7 |
| 55.89% | 76.78% | 45.71% | 556 | 289 | 235 | 54 | 268 | 3 | 3 | 0.7 | 0.8 |
| 55.64% | 87.85% | 41.85% | 556 | 256 | 234 | 22 | 300 | 3 | 3 | 0.7 | 0.9 |
| 65.25% | 60.98% | 80.02% | 556 | 471 | 302 | 169 | 105 | 4 | 2 | 0.4 | 0.7 |
| 66.23% | 69.02% | 70.72% | 556 | 415 | 301 | 114 | 150 | 4 | 2 | 0.4 | 0.8 |
| 66.62% | 87.15% | 55.52% | 556 | 324 | 285 | 39 | 232 | 4 | 2 | 0.4 | 0.9 |
| 64.66% | 65.51% | 72.12% | 556 | 421 | 291 | 130 | 145 | 4 | 2 | 0.5 | 0.7 |
| 64.74% | 72.01% | 65.25% | 556 | 381 | 289 | 92 | 181 | 4 | 2 | 0.5 | 0.8 |
| 64.50% | 90.16% | 52.07% | 556 | 298 | 270 | 28 | 258 | 4 | 2 | 0.5 | 0.9 |
| 61.94% | 71.05% | 61.55% | 556 | 359 | 268 | 91 | 202 | 4 | 2 | 0.6 | 0.7 |
| 61.80% | 76.11% | 58.16% | 556 | 334 | 266 | 68 | 226 | 4 | 2 | 0.6 | 0.8 |
| 62.13% | 92.19% | 48.54% | 556 | 274 | 253 | 21 | 282 | 4 | 2 | 0.6 | 0.9 |
| 54.60% | 75.39% | 44.04% | 556 | 284 | 229 | 55 | 273 | 4 | 2 | 0.7 | 0.7 |
| 54.34% | 79.74% | 42.25% | 556 | 269 | 227 | 42 | 288 | 4 | 2 | 0.7 | 0.8 |
| 53.48% | 89.69% | 39.06% | 556 | 239 | 223 | 16 | 317 | 4 | 2 | 0.7 | 0.9 |
| 65.25% | 60.98% | 80.02% | 556 | 471 | 302 | 169 | 105 | 4 | 3 | 0.4 | 0.7 |
| 66.23% | 69.02% | 70.72% | 556 | 415 | 301 | 114 | 150 | 4 | 3 | 0.4 | 0.8 |
| 66.62% | 87.15% | 55.52% | 556 | 324 | 285 | 39 | 232 | 4 | 3 | 0.4 | 0.9 |
| 64.66% | 65.51% | 72.12% | 556 | 421 | 291 | 130 | 145 | 4 | 3 | 0.5 | 0.7 |
| 64.74% | 72.01% | 65.25% | 556 | 381 | 289 | 92 | 181 | 4 | 3 | 0.5 | 0.8 |
| 64.50% | 90.16% | 52.07% | 556 | 298 | 270 | 28 | 258 | 4 | 3 | 0.5 | 0.9 |
| 61.94% | 71.05% | 61.55% | 556 | 359 | 268 | 91 | 202 | 4 | 3 | 0.6 | 0.7 |
| 61.80% | 76.11% | 58.16% | 556 | 334 | 266 | 68 | 226 | 4 | 3 | 0.6 | 0.8 |
| 62.13% | 92.19% | 48.54% | 556 | 274 | 253 | 21 | 282 | 4 | 3 | 0.6 | 0.9 |
| 54.60% | 75.39% | 44.04% | 556 | 284 | 229 | 55 | 273 | 4 | 3 | 0.7 | 0.7 |
| 54.34% | 79.74% | 42.25% | 556 | 269 | 227 | 42 | 288 | 4 | 3 | 0.7 | 0.8 |
| 53.48% | 89.69% | 39.06% | 556 | 239 | 223 | 16 | 317 | 4 | 3 | 0.7 | 0.9 |

Opaque Colony Count Results for the Process of Kis et al. [9]

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| 12442.jpg | 12444.jpg | 12452.jpg | 12454.jpg |
|  |  |  |  |
| 12455.jpg | 12456.jpg | 12457.jpg | 12460.jpg |
|  |  |  |  |
| 12461.jpg | 12463.jpg | 12465.jpg | 12466.jpg |
|  |  |  |  |
| 12470.jpg | 12471.jpg | 12475.jpg | 12476.jpg |
|  |  |  |  |
| 12478.jpg | 12479.jpg | 12480.jpg | 12481.jpg |
|  |  |  |  |
| 12483.jpg | 12489.jpg | 12490.jpg | 12492.jpg |
|  |  |  |  |
| 12495.jpg | 12497.jpg | 12500.jpg | 12503.jpg |
|  |  |  |  |
| 12505.jpg | 12507.jpg |  |  |

Opaque Colony Count Results for the modified Circular Hough Transform.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| 12442.jpg | 12444.jpg | 12452.jpg | 12454.jpg |
|  |  |  |  |
| 12455.jpg | 12456.jpg | 12457.jpg | 12460.jpg |
|  |  | A picture containing gauge  Description automatically generated | A picture containing indoor  Description automatically generated |
| 12461.jpg | 12463.jpg | 12465.jpg | 12466.jpg |
|  |  |  |  |
| 12470.jpg | 12471.jpg | 12475.jpg | 12476.jpg |
|  |  |  |  |
| 12478.jpg | 12479.jpg | 12480.jpg | 12481.jpg |
|  | A picture containing indoor, gauge, device  Description automatically generated |  |  |
| 12483.jpg | 12489.jpg | 12490.jpg | 12492.jpg |
|  |  | A picture containing indoor, gauge  Description automatically generated | A picture containing indoor, white  Description automatically generated |
| 12495.jpg | 12497.jpg | 12500.jpg | 12503.jpg |
|  |  |  |  |
| 12505.jpg | 12507.jpg |  |  |

Translucent Colony Count Results for the modified Circular Hough Transform.

|  |  |  |  |
| --- | --- | --- | --- |
| A picture containing indoor, gauge, device, image  Description automatically generated |  |  | A circular white object with writing on it  Description automatically generated with low confidence |
| 12442.jpg | 12444.jpg | 12452.jpg | 12454.jpg |
|  | A picture containing round, gauge, dark  Description automatically generated | A picture containing indoor, gauge  Description automatically generated | A picture containing indoor, gauge  Description automatically generated |
| 12455.jpg | 12456.jpg | 12457.jpg | 12460.jpg |
| A picture containing indoor, green, device, gauge  Description automatically generated | A picture containing indoor, gauge, device  Description automatically generated | A picture containing indoor, Petri dish, green  Description automatically generated |  |
| 12461.jpg | 12463.jpg | 12465.jpg | 12466.jpg |
| A picture containing green, device  Description automatically generated | A picture containing indoor, device, gauge  Description automatically generated |  |  |
| 12470.jpg | 12471.jpg | 12475.jpg | 12476.jpg |
| A picture containing indoor, device, gauge  Description automatically generated | A picture containing indoor  Description automatically generated | A picture containing indoor  Description automatically generated |  |
| 12478.jpg | 12479.jpg | 12480.jpg | 12481.jpg |
| A picture containing Petri dish  Description automatically generated |  |  |  |
| 12483.jpg | 12489.jpg | 12490.jpg | 12492.jpg |
|  |  |  |  |
| 12495.jpg | 12497.jpg | 12500.jpg | 12503.jpg |
|  | A picture containing indoor, green  Description automatically generated |  |  |
| 12505.jpg | 12507.jpg |  |  |

Opaque Colony Count Results for Blob Detection.

|  |  |  |  |
| --- | --- | --- | --- |
| A picture containing indoor, black, gauge  Description automatically generated | A picture containing indoor, Petri dish, gauge  Description automatically generated |  |  |
| 12442.jpg | 12444.jpg | 12452.jpg | 12454.jpg |
| A picture containing indoor, different, gauge  Description automatically generated |  | A picture containing indoor, gauge  Description automatically generated |  |
| 12455.jpg | 12456.jpg | 12457.jpg | 12460.jpg |
|  |  |  |  |
| 12461.jpg | 12463.jpg | 12465.jpg | 12466.jpg |
|  |  |  |  |
| 12470.jpg | 12471.jpg | 12475.jpg | 12476.jpg |
|  |  | A picture containing indoor  Description automatically generated |  |
| 12478.jpg | 12479.jpg | 12480.jpg | 12481.jpg |
| A picture containing Petri dish, tableware  Description automatically generated | A round white object with writing on it  Description automatically generated with low confidence |  |  |
| 12483.jpg | 12489.jpg | 12490.jpg | 12492.jpg |
|  |  |  | A picture containing indoor  Description automatically generated |
| 12495.jpg | 12497.jpg | 12500.jpg | 12503.jpg |
|  |  |  |  |
| 12505.jpg | 12507.jpg |  |  |

Translucent Colony Count Results for Blob Detection.

|  |  |  |  |
| --- | --- | --- | --- |
|  | A picture containing gauge, device  Description automatically generated |  | A picture containing indoor, gauge, round  Description automatically generated |
| 12442.jpg | 12444.jpg | 12452.jpg | 12454.jpg |
| **A picture containing indoor, Petri dish  Description automatically generated** |  | A picture containing indoor, gauge  Description automatically generated |  |
| 12455.jpg | 12456.jpg | 12457.jpg | 12460.jpg |
|  |  | A picture containing indoor, gauge, device  Description automatically generated |  |
| 12461.jpg | 12463.jpg | 12465.jpg | 12466.jpg |
|  | A picture containing indoor, gauge, device  Description automatically generated |  |  |
| 12470.jpg | 12471.jpg | 12475.jpg | 12476.jpg |
|  |  | A picture containing indoor, Petri dish, tableware  Description automatically generated |  |
| 12478.jpg | 12479.jpg | 12480.jpg | 12481.jpg |
|  |  |  |  |
| 12483.jpg | 12489.jpg | 12490.jpg | 12492.jpg |
|  |  |  |  |
| 12495.jpg | 12497.jpg | 12500.jpg | 12503.jpg |
|  |  |  |  |
| 12505.jpg | 12507.jpg |  |  |

# References

1. Anon. Bacteria: Types, characteristics, where they live, hazards, and more. Retrieved September 26, 2021 from https://www.medicalnewstoday.com/articles/157973#structure
2. Anon. What is computer vision? Retrieved September 26, 2021 from

https://www.ibm.com/topics/computer-vision

1. Pal, N., Swadi, N., Agarwal, P., Prajval, & Goel, A. (2021, January 4). Computer Vision in Automotive Industry – A Guide. CronJ. Retrieved October 10, 2021, from https://www.cronj.com/blog/computer-vision-in-automotive-industry-a-guide/.
2. Fingerprint, P. (2021, February 16). Computer vision in agriculture. Medium. Retrieved October 10, 2021, from https://medium.com/swlh/computer-vision-in-agricultured84b69c6858e.
3. Microbiology Society. Bacteria: What is microbiology? Retrieved September 26, 2021 from https://microbiologysociety.org/why-microbiology-matters/what-ismicrobiology/bacteria.html
4. Lola Hogekamp, Stefan H. Hogekamp, and Mario R. Stahl. 2020. Experimental setup and image processing method for automatic enumeration of bacterial colonies on agar plates. *PLOS ONE* 15, 6 (2020). DOI:http://dx.doi.org/10.1371/journal.pone.0232869
5. Daniela A. García-Soriano, Frederikke Dybdahl Andersen, Jens Vinge Nygaard, and Thomas Tørring. 2021. ColFeatures: Automated data extraction and classification of bacterial colonies. (2021). DOI:http://dx.doi.org/10.1101/2021.05.27.445853
6. Thomas Beznik, Paul Smyth, Gaël de Lannoy, and John A. Lee. 2020. Deep learning to detect bacterial colonies for the production of vaccines. (September 2020). Retrieved September 10, 2021 from https://arxiv.org/abs/2009.00926
7. Busra Kis, Mazlum Unay, Gizem Dilara Ekimci, Utku Kursat Ercan, and Aydin Akan. 2019. Counting bacteria Colonies based on image processing methods. *2019 Medical Technologies Congress (TIPTEKNO)* (2019). DOI:http://dx.doi.org/10.1109/tiptekno.2019.8895213
8. Marquard, D., Austerjost, J., Dusterloh, S., Barthold, C., Beutel, S., Scheper, T., and Lindner, P. 2018. A novel image analyzing algorithm for colony plate counting with handheld devices. *Advances in Biotechnology & Microbiology* 11, 1 (2018). DOI:http://dx.doi.org/10.19080/aibm.2018.11.555804
9. Lei Huang and Tong Wu. 2018. Novel neural network application for bacterial colony classification. *Theoretical Biology and Medical Modelling* 15, 1 (2018). DOI:http://dx.doi.org/10.1186/s12976-018-0093-x
10. Muhammed Talo. 2019. An automated deep learning approach for bacterial image classification. (December 2019). Retrieved September 10, 2021 from

https://arxiv.org/abs/1912.08765

1. Nicholas Stolze, Carly Bader, Christopher Henning, Jared Mastin, Andrea E. Holmes, and Arin L. Sutlief. 2019. Automated image analysis with ImageJ of yeast colony forming units from cannabis flowers. *Journal of Microbiological Methods* 164 (2019), 105681. DOI:http://dx.doi.org/10.1016/j.mimet.2019.105681
2. Volkan Altuntas, Seda Altuntas, M. Gok. 2018. Automatic colony segmentation on agar surface by image processing. *Journal of Scientific & Industrial Research* 77, pp. 451-456 (2018). Retrieved from http://nopr.niscair.res.in/bitstream/123456789/44804/1/JSIR%2077(8)%20451-456.pdf
3. Paolo Andreini, Simone Bonechi, Monica Bianchini, Alessandro Mecocci, and Franco Scarselli. 2018. A Deep Learning Approach to Bacterial Colony Segmentation. (August 2018). Retrieved September 26, 2021 from doi:10.1007/978-3-030-014247\_51
4. Shi, J., Zhang, F., Wu, S., Guo, Z., Huang, X., Hu, X., Holmes, M. and Zou, X., 2019. Noise- free microbial colony counting method based on hyperspectral features of agar plates. Food chemistry, 274, pp.925-932. doi:10.1016/j.foodchem.2018.09.058
5. Wong, C.F., Yeo, J.Y. and Gan, S.K.E., 2019. Republication–APD Colony Counter App: Using Watershed Algorithm for improved colony counting. Sci. Phone Apps Mob. Devices, 5(5).
6. Lin, H.Y., Chen, S.Y., Chung, C.L. and Kuo, Y.F., 2019. Counting bacterial colony on agar plates using deep convolutional neural network. In 2019 ASABE Annual International Meeting (p. 1). American Society of Agricultural and Biological Engineers. doi:10.13031/aim.201900485
7. Zhu, G., Yan, B., Xing, M. and Tian, C., 2018. Automated counting of bacterial colonies on agar plates based on images captured at near-infrared light. Journal of microbiological methods, 153, pp.66-73.. doi:10.1016/j.mimet.2018.09.004
8. Badieyan, S., Dilmaghani-Marand, A., Hajipour, M. J., Ameri, A., Razzaghi, M. R., RafiiTabar, H., ... & Sasanpour, P., 2018. Detection and discrimination of bacterial colonies with Mueller matrix imaging. Scientific reports, 8(1), 1-10.
9. Wang, H., Koydemir, H. C., Qiu, Y., Bai, B., Zhang, Y., Jin, Y., ... & Ozcan, A., 2020. Early detection and classification of live bacteria using time-lapse coherent imaging and deep learning. Light: Science & Applications, 9(1), 1-17.
10. Rao, A. S., Goveas, L. C., & Nayak, S., 2021. Swift Automated System for Distinguishing Blue-White Colonies Post Bacterial Transformation on Agar Plates Using Computer Vision Techniques. Philippine Journal of Science, 150(5), 933-942.
11. Lee, A., Park, S., Yoo, J., Kang, J., Lim, J., Seo, Y., ... & Kim, G., 2021. Detecting Bacterial Biofilms Using Fluorescence Hyperspectral Imaging and Various Discriminant Analyses. Sensors, 21(6), 2213.
12. Treebupachatsakul, T., & Poomrittigul, S., 2019. Bacteria classification using image processing and deep learning. In 2019 34th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC) (pp. 1-3). IEEE.
13. Lecky, F. (2001). Figure 1. ResearchGate. Retrieved October 30, 2021, from

https://www.researchgate.net/figure/Extract-of-the-t-table-The-first-column-lists-thedegrees-of-freedom-n-1-The\_fig1\_12025083.

1. Hamdani, I. 2020. Counting of Bacterial Colonies of the Low Quality Image Using Perona-Malik Diffusion Filters and Image Morphology Operators. Retrieved Oc, (2020) Retrieved from <https://iopscience.iop.org/article/10.1088/1742-6596/1490/1/012058>
2. Jiyun Byun, Mark R. Verardo, Baris Sumengen, Geoffrey P. Lewis, B.S. Manjunath, and Steven K. Fisher. 2006. Automated tool for the detection of cell nuclei in digital microscopic images: Application to retinal images. *Molecular Vision* 12 (Aug. 2006). PMID: 16943767.
3. Lei Yang, Richard Parton, Graeme Ball, Zhen Qiu, Alan H. Greenaway, Ilan Davis, and Weiping Lu. 2010. An adaptive non-local means filter for denoising live-cell images and improving particle detection. *J Struct Biol* 172, 3 (Dec. 2010). DOI: 10.1016/j.jsb.2010.06.019
4. Grossi M., Parolin C., Vitali B., and Riccò B. 2019. Computer Vision Approach for the Determination of Microbial Concentration and Growth Kinetics Using a Low Cost Sensor System. *Sensors (Basel)* 19, 24 (Dec. 2019). DOI: 10.3390/s19245367
5. Andrea Loddo, Cecilia Di Ruberto, and Giovanni Puglisi. 2020. Blob detection and deep learning for leukemic blood image analysis. *Appl Sci.* 10, 3 (Feb. 2020). DOI: 10.3390/app10031176
6. Min Zhang, Teresa Wu, Kevin M. Bennett. 2014. Small blob identification in medical images using regional features from optimum scale.
7. Yuan-Hsiang Chang, Lara A. Hardesty, Christiane M. Hakim, Thomas S. Chang, Bin Zheng, Walter F. Good, David Gur. 2001. Knowledge-based computer-aided detection of masses on digitized mammograms: A preliminary assessment