CS591 Project Report

Counting distinct objects from an image using Image Processing and Analysis Techniques

Submitted in partial fulfilment of the requirements
for the award of the degree of

Bachelor of Technology
in
Computer Science and Engineering

Submitted by

Devadi Yekaditya (CSE/20039/559)
Mislah Rahman CP (CSE/20055/575)
Sriramsetty Bhanu Teja (CSE/20088/608)

Under the guidance of
Dr. Amit Ranjan Azad

Indian Institute of Information Technology Kalyani
West Bengal, India, 741235
Autumn 2022

Contents

Indian Institute of Information Technology Kalyani

Certificate

This is to certify that the thesis entitled **“Counting distinct objects from an image using Image Processing and Analysis Techniques”** being submitted by Devadi Yekaditya (CSE/20039/559), Mislah Rahman CP (CSE/20055/575), Sriramsetty Bhanu Teja (CSE/20088/608), undergraduate students of Indian Institute of Information Technology Kalyani, West Bengal, 741235, India, for the award of Bachelors of Technology in Computer Science and Engineering is an original research work carried by them under my supervision and guidance. The thesis has fulfilled all the requirements as per the regulations of Indian Institute of Information Technology Kalyani and in my opinion, has reached the standards needed for submission. The works, techniques and the results presented have not been submitted to any other University or Institute for the award of any other degree or diploma.

Dr. Amit Ranjan Azad
Assistant Professor

Date:

Declaration

We hereby declare that the work which is being presented in the thesis entitled **“Counting distinct objects from an image using Image Processing and Analysis Techniques”** is submitted to the Indian Institute of Information Technology Kalyani in partial fulfillment for the award of the degree of Bachelor of Technology in Computer Science and Engineering during the period from July to November, 2022 under the supervision of Dr. Amit Ranjan Azad, Indian Institute of Information Technology Kalyani, West Bengal 741235, India, does not contain any classified information.

Devadi Yekaditya (CSE/20039/559)
Mislah Rahman CP (CSE/20055/575)
Sriramsetty Bhanu Teja (CSE/20088/608)

Computer Science and Engineering
Indian Institute of Information Technology Kalyani

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

-------------------
Dr. Amit Ranjan Azad
Assistant Professor
Indian Institute of Information Technology Kalyani

-------------------
Place: Kalyani, West Bengal
Date:

Acknowledgement

We would like to thank our supervisor Dr. Amit Ranjan Azad for giving us the opportunity for working on this project entitled **“Counting distinct objects from an image using Image Processing and Analysis Techniques”.** His constant guidance and useful insights led the development of this work and is due to his support that the work can be submitted in present state.

Devadi Yekaditya (CSE/20039/559)
Mislah Rahman CP (CSE/20055/575)
Sriramsetty Bhanu Teja (CSE/20088/608)

Computer Science and Engineering
Indian Institute of Information Technology Kalyani

Place: Kalyani, West Bengal
Date:

Abstract

This report summarises the usage of image processing and analysis techniques to count descriptive, distinct, non derogative and non redundant objects in an image. The work will be helpful in automating various tasks that primarily requires to count objects. It’s very difficult for human to keep track of objects while counting since our short term memory is very limited and we often end up in a situation that if we have counted the particular object before or not. The project long to automate deceptively silly but labour intensive task of counting distinct object in real world scenarios. The project is intended to be scaled for larger applications by incorporating deep learning based techniques to provide more accurate and consistent results.

Introduction

We often stumble at situations at which we have to count distinct objects from a collection of objects or from an image. It may seem silly at a glance but once someone start to counting the number of objects, it gets increasingly difficult to keep track of the objects which are already been counted. While counting we may skip some counts or repeat counting the same object unknowingly. This is because our short-term memory is too limited to store which all objects we have already counted and our brain is not at all efficient in focusing multiple tasks at once (here counting and keeping track of objects).

Counting distinct object have applications in all fields of human life. From science, technology, engineering to medical, agricultural and even in day-to-day life. We require counting a number of different things for various reasons, ranging from number of visible stars at night sky to number of sheep in a farm to number of bacteria in a microscopic image.

Various manual counting techniques are devised throughout the history out of which, the easiest way is to use a tally counter to keep track of the numbers, so that the brain can focus on the objects. Another approach is to mark the objects which have been counted, so that the brain can focus on the count. By combining both approaches, we can achieve fairly accurate results despite the large amount of time being consumed. Other manual techniques include arranging the objects into points on a rectangle so that the area is the number of objects, or grouping objects into small equal groups, or taking a picture and drawing line on the objects while counting, etc.

Since the advent of industrialization, most of the labor-oriented tasks are taken by machines. Several mechanical and electrical techniques are discovered to do the task of counting objects. After the arrival of general-purpose computers and its requirement in science and navigation, the importance of the field, image processing and analysis have gained its momentum. Now in the age of cheap credit card sized computers and inexpensive high-quality cameras, it become increasingly obsolete to use the classical approaches to count objects.

Even though there exist a number of algorithms and techniques to accurately count the number of distinct objects in an image, we decided to try our own approach in doing the same. We are using python to implement our work and are using the following technologies:

Technology Used: NumPy

Python doesn’t natively support C like arrays. The inbuild list is too slow for processing large datasets. NumPy array provides efficient C like arrays, moreover large collection of complex matrix manipulations, arithmetic, statistical and other high level mathematical operations are available to operate on these arrays.

Typical images are 24bit array of pixels in RGB color space. They are usually compressed in various formats and standards and can even be of different number of bits (depth). Each pixel is further divided into three 8bit subpixels on which the intensity of the primary additive colors Red, Green and Blue are stored in order.

A standard high definition 24bit image of dimension 1280 x 720 is represented in a 1280 x 720 x 3 array with 2764800 elements in it. The size of an image itself justify the requirement of NumPy arrays.

Technology Used: OpenCV

OpenCV is an abbreviation of the term Open Source Computer Vision Library. OpenCV is a library of various algorithms and techniques used for image processing and analysis. OpenCV is written in C++ and allows GPU based hardware accelaration. Eventhough primarily it interface with C++, there are API support for major programming languages like C, python, java and matlab.

The Python version of OpenCV uses NumPy arrays as primary data type to store and process data. OpenCV doesn't acquire NumPy object to create a OpenCV object as most of the other libraries does. Instead, it directly process on NumPy arrays and keep the array object properties as it is. The Merit of this is that all NumPy methods and techniques can be used without converting between object types.

Eventhough modern standard for physical color space is RGB, OpenCV uses BGR color space instead due to historical reasons, and it is now virtually impossible as well trivial to make it use RGB color space upon considering the vast amount of libraries, applications and softwares written using OpenCV.

Work Flow

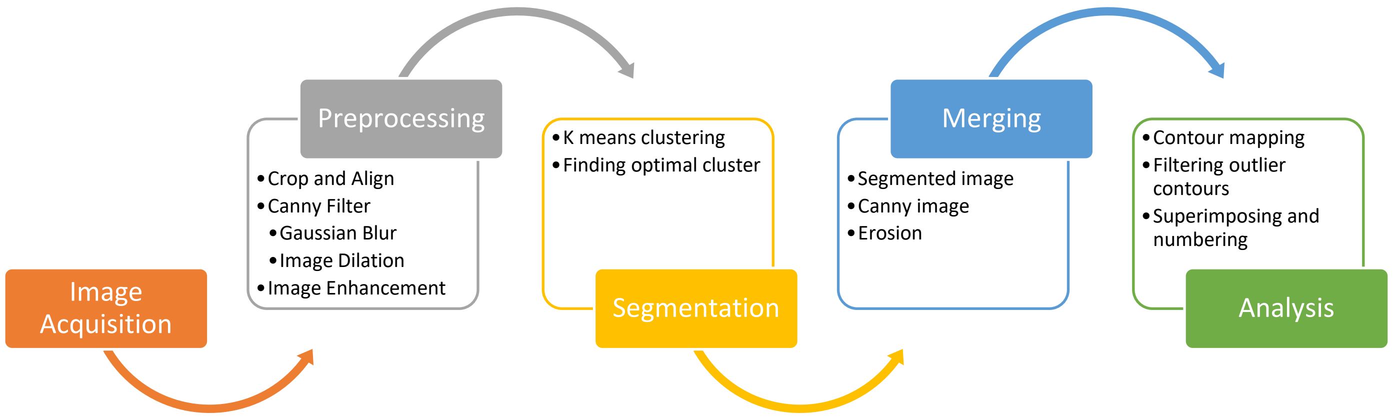


Image Acquisition & Pre-Processing

Image Acquisition

Image should be of good quality for accurately analysing the image. Getting an image through a high quality camera is recommended. The image should have optimal lighting conditions and the subject should be visually distinguishable from the backgrounds for achieving expected results.

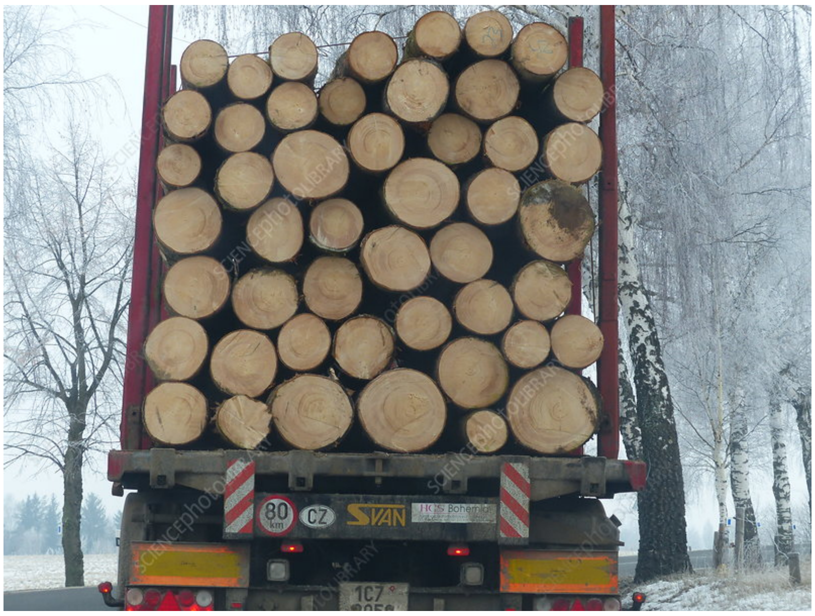


Figure 1: Image obtained from camera

Crop & Align Image

The image contains areas with unnecessary extra area round the subject which are out of context and makes the counting of object more complex. Inorder to prevent the wastage of computation and optimize the counting algorithm, the image should be trimmed such that the it the edges align with edges of the subject. To maximise the removal of unnecessary background, image may be rotated before cropping.

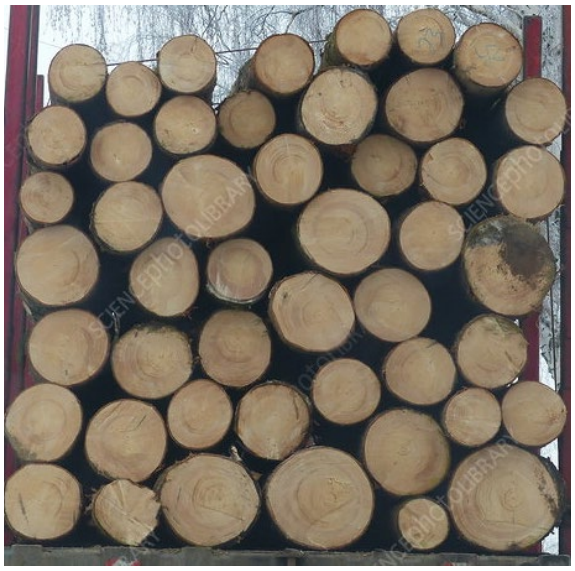


Figure 2: Image after cropping

Edge Detection (Canny edge detection)

Canny edge detection algorithm is used to detect the edge of the image. The canny edge algorithm can't be used alone to count the object since it usually contains a lot of noise. The edge detection algorithm follows the following steps to detect edges:

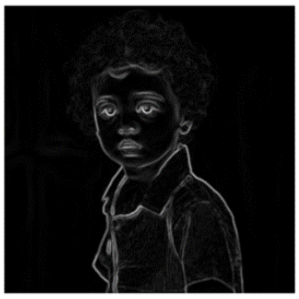
i. Noise reduction using Gaussian blurring

Greyscale

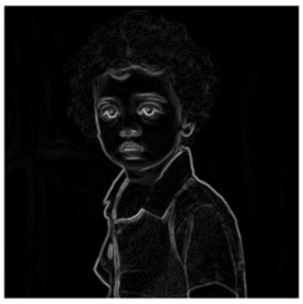
Blurred

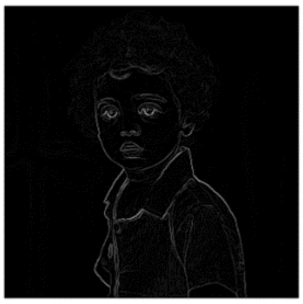
ii. Obtaining gradient magnitude

Blurred

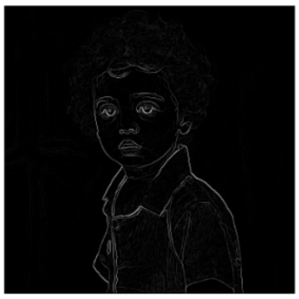

Gradient Magnitude

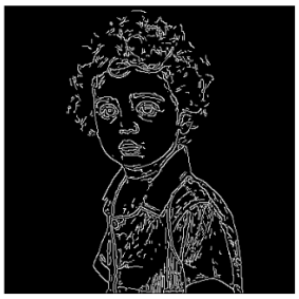
iii. Non-max Suppression


Gradient Magnitude


Non-max Suppression

iv. Hysteresis Thresholding


Non-max Suppression


Final Output

A gaussian blur of kernel size 3x3 is applied to the cropped image before applying canny filter to reduce the noise in the image.

Image Dilation

Image dilation increases the boundary thickness by growing boundary. After canny edge detection, the edge seems to be too narrow. So, adding some extra pixels to the boundary so that the edge is easily visible. Doing so also helps in smoothening sharp curves.

Our image is dilated with a kernal matrix of kernal size 2x2 with two iterations. The image is further inverted to match the background for using it later.

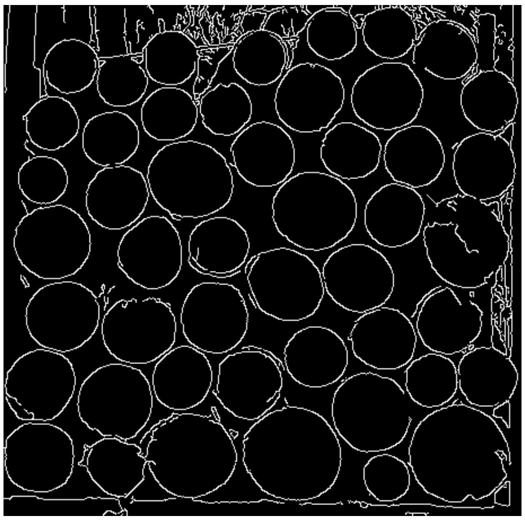


Figure 3: Canny edges

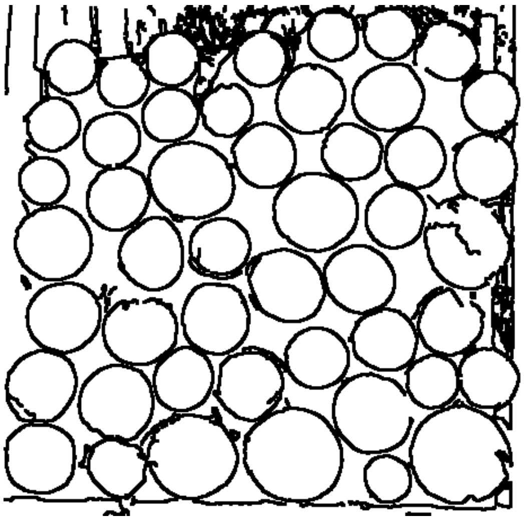


Figure 4: Inverted canny edge after dilation

Image Enhancement

Now going back to the partially preprocessed image by cropping and realignment. The image shall be enhanced to make it's key features easily be visually distinguishable. The cropped image is enhanced by applying the techniques provided in the following section.

Converting from BGR to HSV

In BGR, each of the pixel holds the intensity value which is correlated to the color luminance of blue, green and red channel. So, converting to HSV (Hue, Saturation and Value) color space makes it easier to adjust the luminance and saturation. Since color information can be stored in a single byte and the remaining two bytes are used to store Saturation and Luminance.

The saturation is increased by 450%, so that the background and foreground look fairly distinct even when the color is almost same but different. The value is also increased by 150%, so that the brighter areas of the image appear more brighter, hence it makes the features easily distinguishable.

After increasing the saturation and value, the pixels are trimmed back to 24bit color space and converted back to bgr color space.

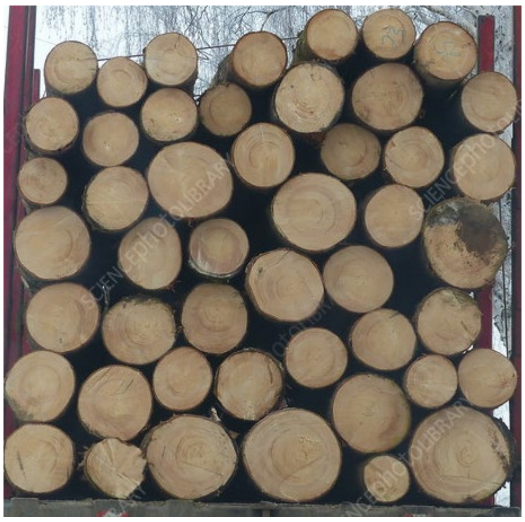


Figure 5: Cropped image

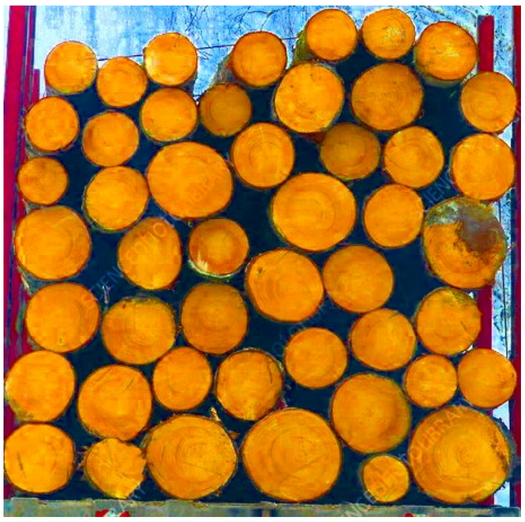


Figure 6: Enhanced image

Segmentation (K-Means Clustering)

K means clustering

K means clustering is an unsupervised learning algorithm used to group unlabeled data into K number of clusters out of which each data point belongs to the cluster with nearest mean distance.

K means clustering with cluster count, k = 3 is used here to filter out the subject from the background, since the real world images contains an immediate secondary background around the subject distinct from the primary background.

If it is clear that there doesn’t exist a secondary background, k shall be substituted with 2, especially in case of digital graphics.

Finding optimal cluster:

Since K means clustering with 3 means have done, there must be 3 segments and it is required to separate the subject out of them. It is particularly required that the contour mapping technique which is used for analysis works best with binary images. So, the primary and the secondary background to be identified and merge them to a single background segment and the subject to be kept in the foreground. This step can be destructive since, error in guessing the subject accurately will make the subject fall in background and make it impossible to count.

Since the image is expected to be cropped by the user, we can safely assume that the foreground of the image covers maximum area in most of the cases (it can be better adjusted depending the type of the image), Now the segments with least number of pixels, that is the segments with least area are merged together to form the background.

If the image is not cropped, then different approaches have to be implemented to effectively deal the situation. If the mean of euclidian distances of pixels of a cluster from it's cluster center is very close to the image center and the standard deviation is very low, then it can be considered as foreground.

As the maximum number of pixels are occupied by the subject, here the segments with least number of pixels are merged to form a binary image and is further recolored.

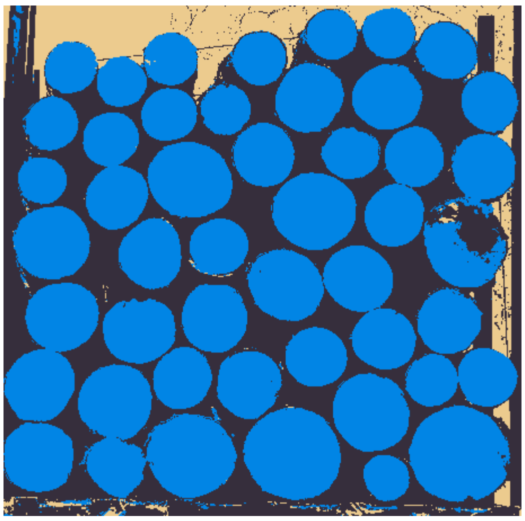


Figure 7: Segmented image with k = 3

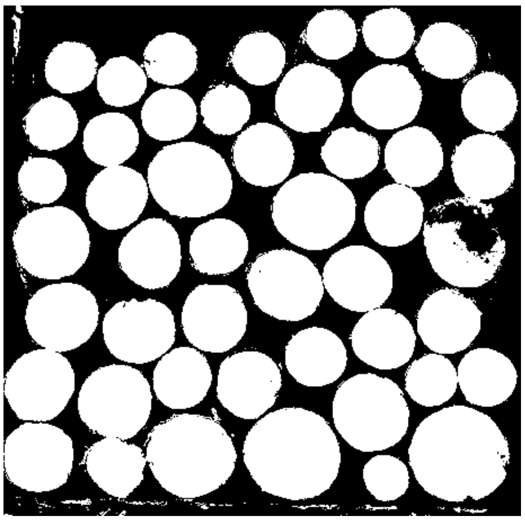


Figure 8: After selecting optimal segment

Merging

The segmentation using K means clustering may erode some of the edges, which affects the contour mapping and finally the object counting itself. By superimposing canny filtered image over the segmented image, the losing of edges can be prevented.

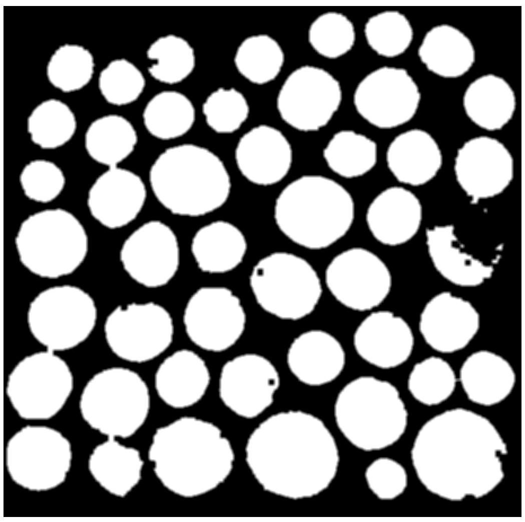


Figure 3: Segmented image after gaussian filtering and erosion

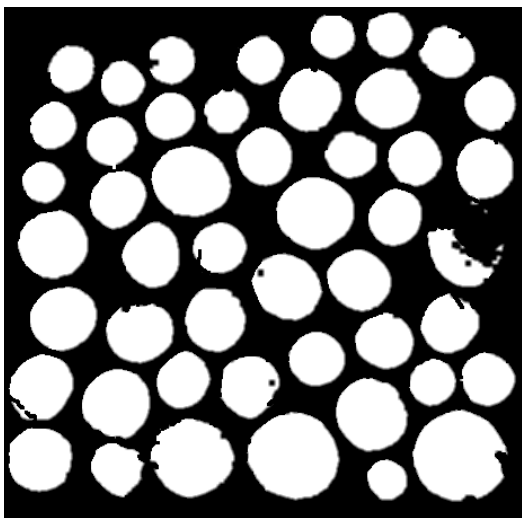


Figure 4: Segmented image after superimposing with canny edge

Contour Mapping & Distinct object counting

Contours are simple curves joining all continuous points with same color and intensity. Contour function of OpenCV returns the points around the object. Also, the nested objects detected are classified to hierarchies. To count the objects, the best is to count number or labels in the first level hierarchy detected by the contour function of OpenCV. The object with low area and has very high standard deviation with other objects might be noise or other random objects. By assuming that the area of contours follows normal distribution, The outlier points with area x, $x < \bar{x} - 2 \cdot \sigma$, are removed.

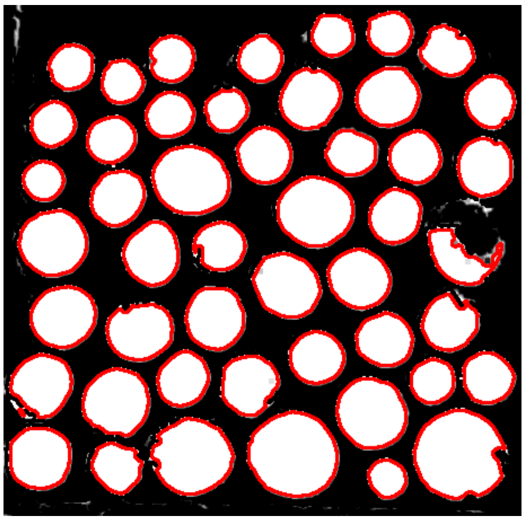


Figure 61: After drawing bounding boxes around first level contours

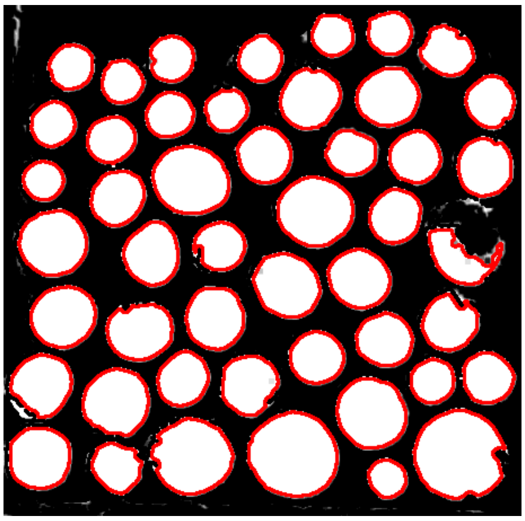


Figure 5: After removing outlier points

The centers of each contour are found and are marked with their index. The contours found are superimposed on original image with the markings and their numbers. The contour and text color are chosen such that it is easily readable by the user. Total count of objects detected is printed on the top left corner.

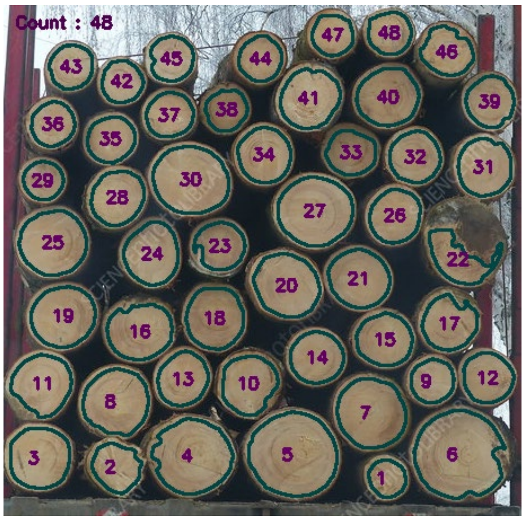


Figure 73: After superimposing on original image and numbering

Results

The program is tested and verified on various images with visually distinguishable objects. So of the tests are documented quantitatively in the following table. The number of distinct objects is manually counted by a human subject and it is compared with the result provided by the program.

The accuracy is calculated as:

$$Accuracy = \frac{Actual\ number - \ difference}{Actual\ number} \times 100\%$$

Serial number of image	Actual number of Objects	Object detected by the program	Difference	Accuracy
1	48	48	0	100.00%
2	59	60	1	98.31%
3	92	94	2	97.83%
4	14	14	0	100.00%
5	25	26	1	96.00%
6	42	42	0	100.00%
7	36	37	1	97.22%
8	65	66	1	98.46%
9	63	61	2	96.83%
10	5	5	0	100.00%

The results are consistent with a mean accuracy of 98.465% and standard deviation 1.496.

Conclusion

Here we successfully achieved a consistent result with accuracy over 95% for the tested images. Which is fairly good for an unoptimized technique when compared to the advanced techniques already available. Since there are chances for error, it can’t be yet used for a fully automated system without human assistance.

The drawing of circles and numbering object helps the user to easily identify any errors and manually fix it. Since the circles are superimposed on the original image, user find it easy to identify any missing circle or extra circle and can fix the count manually.

The report provides a promising approach to count descriptive, distinct, non derogative and non redundant objects in an image based on classical image processing and analysis techniques.

Applications

This work can be used in automating various tasks that primarily requires counting objects ranging from production lines, scientific and engineering applications to various day to day life activities. The scope of the projects includes all domains of human life.

Production lines often requires manual labors for counting objects before final packaging or for counting object in batch production, upon automating this task, the labor cost can be cut down to a great extend and human error can be reduced.

The project can be particularly useful for tedious tasks like counting number of cells or number of bacteria in a microscopic image, counting number of stars etc. which increases efficiency and productivity in various scientific, engineering and medical field applications.

In day-to-day life, we often stumble across various tasks that requires counting object like, number of logs in a truck, number of bricks in a stack, number keys in a keyboard, number of characters in a printed text etc. which can easily be counted using this work. The project finds its application in all fields, for example in agriculture, it can be used to count number of missing sheep from a large farm or can be used to count number of germinations in a tissue culture, etc. Thus, the application scope of this project is virtually unlimited.

Future Scope

The algorithm can include a way to automate cropping and geometric realignment, so that manual task required can be reduced or further eradicated.

A Deep Learning based model can be incorporated with the classical algorithm, so that it can detect wider range of objects and provide better accuracy.

Presently the project is using K means segmentations based on color values, so the algorithm gives inaccurate result if objects to be counted are of different colors, The algorithm can be upgraded to heuristically select from color, texture and structural similarity to get optimal result.

The project can be implemented to an android app with real time counting so that it will be more useful end users.

Bibliography & References:

- Digital Image Processing, 3rd edition - Rafael C. Gonzales, Richard E. Woods
- Digital Image Processing and Analysis, 2nd edition – B. Chanda and Dutta Majumder; PHI Learning.
- Yella, Siril and Dougherty, Mark. "Automatically Detecting the Number of Logs on a Timber Truck" *Journal of Intelligent Systems*, vol. 22, no. 4, 2013, pp. 417-435. https://doi.org/10.1515/jisys-2013-0026