Automatic Fish Egg Counting System using Image Processing Techniques

A thesis submitted in partial fulfilment of the requirement for the

Degree of Master of Technology

Of

Jadavpur University

By

Snehasis Sahu

Registration Number: 149850 of 2019-2020

Examination Roll Number: M6TCT22017

Under the Guidance of

Dr. Nibaran Das

Professor

Department of Computer Science and Engineering

Jadavpur University, Kolkata-700032

India

June, 2022

FACULTY OF ENGINEERING AND TECHNOLOGY

JADAVPUR UNIVERSITY

CERTIFICATE OF RECOMMENDATION

This is to certify that the thesis entitled "**Automatic Fish Egg Counting System using Image Processing Techniques**" has been satisfactorily completed by **Snehasis Sahu** (University Registration No.:149850 of 2019-2020, Exam Roll No.:M6TCT22017). It is a bonafide piece of work carried out under my guidance and supervision and be accepted in partial fulfilment of the requirement for the Degree of **Master of Technology in Computer Technology in Department of Computer Science and Engineering, Faculty of Engineering and Technology, Jadavpur University, Kolkata 700032**.

Dr. Nibaran Das (Thesis Supervisor)
Professor
Department of Computer Science and Engineering,
Jadavpur University, Kolkata-700032.

Countersigned

Prof. Anupam Sinha
Head, Computer Science and Engineering,
Jadavpur University, Kolkata-700032.

Prof. Chandan Mazumdar
Dean, Faculty of Engineering and Technology,
Jadavpur University, Kolkata-700032.

FACULTY OF ENGINEERING AND TECHNOLOGY

JADAVPUR UNIVERSITY

CERTIFICATE OF APPROVAL

This is to certify that the thesis entitled "**Automatic Fish Egg Counting System using Image Processing Techniques**" is a bonafide record of work carried out by Snehasis Sahu in partial fulfilment of the requirements for the award of the degree of Master of Technology in Computer Technology in the Department of Computer Science and Engineering, Jadavpur University during the period of 2019 to 2022. It is understood that by this approval the undersigned do not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn there in but approve the thesis only for the purpose for which it has been submitted.

____

Signature of Examiner
Date:

____

Signature of Supervisor
Date:

FACULTY OF ENGINEERING AND TECHNOLOGY

JADAVPUR UNIVERSITY

DECLARATION OF ORIGINALITY AND COMPLIANCE OF ACADEMIC ETHICS

I hereby declare that this thesis entitled "**Automatic Fish Egg Counting System using Image Processing Techniques**" contains literature survey and original research work by the undersigned candidate, as part of his Degree of Master of Technology in Computer Technology.
All information in this document has been obtained and presented in accordance with academic rules and ethical conduct.
I also declare that, as required by these rules and conduct, I have fully cited and referenced all materials and results that are not original to this work.

Name: Snehasis Sahu
University Registration No. : 149850 of 2019-2020
Examination Roll No. : M6TCT22017

Thesis Title: Automatic Fish Egg Counting System using Image Processing Techniques

Signature:
____

Date:
____

ACKNOWLEDGEMENT

First and foremost, I would like to start by thanking God Almighty for showering me with the strength, knowledge and potential to embark on this wonderful journey and to persevere and complete the embodied research work satisfactorily.
I am pleased to express my deepest gratitude to my thesis guide, **Dr. Nibaran Das**, Computer Science and Engineering, Jadavpur University, Kolkata for his invaluable guidance, constant encouragement and inspiration during the period of my dissertation.
I am highly indebted to **Jadavpur University** for providing me the opportunity and the required infrastructure to carry on my thesis.
I am also grateful to the **Center for Microprocessor Applications for Training Education and Research Lab, Computer Science and Engineering department, Jadavpur University** for giving me the proper laboratory facilities as and when required.
I am thankful to all the teaching and non-teaching staff whose helping hands have smoothed my journey through the period of my research.
Last but not the least; I would like to thank my family members, classmates, seniors and friends for giving me constant encouragement and mental support throughout my work.

Snehasis Sahu
University Registration No. : 149850 of 2019-2020
Examination Roll No. : M6TCT22017
Master of Technology
Department of Computer Science and Engineering
Jadavpur University

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

CHAPTER 1

INTRODUCTION

The fishery industry plays a significant role in India’s economy. It serves as a source of income and employment for a large number of coastal fishermen. The country ranks third in fisheries production and second in aquaculture [6]. According to a recent estimate by the National Fisheries Development Board, the fishery industry alone employs 145 million people, contributes 1.07 percent to the nation’s gross domestic product, and generates export earnings of Rs 334.41 billion. According to the National Institute of Agricultural Economics and Policy Research, the expected demand for this financial year will increase to 11.80 million metric tonnes [6].

Fish breeding is an important aspect of the fisheries industry. It is a complicated procedure that requires careful planning and monitoring. The most important aspect of fish breeding is to maintain the optimal egg quality. Estimating the 'spawning biomass' of the adult population during the reproductive season is an important method for assessing the size of fish stocks. This is done by figuring out the female’s relative fecundity, or how many eggs she produces per unit of body weight, and then counting the number of eggs she releases on the spawning grounds. Counting the quantity of fish eggs is essential in determining a variety of fish factors (such as survival rate, reproduction rate, species of fish, and so on). However, counting the quantity of fish eggs is a time-consuming and challenging task. Fish eggs are translucent and normally float on top of the water. People usually put fish eggs on a dish and count them one by one or use a magnifying glass to figure out how many there are. However, this technique takes too long, and the results of manual counting are not always accurate. Researchers have been trying to come up with an automated way to deal with this problem for decades.

The target of this research is to accurately segment and identify eggs from images of eggs placed on a disk. Each sample of the fish eggs was placed into a transparent dish with a 10 mm depth of water. We have dispersed eggs in the dish. We have captured the image from the top side of the dish using a smartphone camera. Then we apply different image processing techniques (such as opening, dilation of image, segmentation, connected components) to identify and count the eggs present in the dish.

CHAPTER 2

LITERATURE SURVEY

For decades, researchers have been attempting to develop an automated method to count the eggs of fish. It is a challenging problem. Several automated methods for counting eggs are described in [3][16]. Counting of fish eggs during reproduction is very important in the fishery industry as described in [2].

Before the introduction of fish egg count machine by Boyar et al. [3] in the year 1967, people used conventional volumetric and gravimetric methods for the counting the eggs. Boyar et al. [3] developed a machine that records dry eggs as they pass a photoelectric cell. In [16], Witthames and Walker proposed an automated process for counting the eggs of fish. Eggs presented in water are pumped at a specific rate through an electronic sensor, which generates a voltage pulse whose amplitude is proportional to particle size. Then they counted the number of particles.

Present-day researchers are more moved to image processing rather than building separate machines for counting the number of eggs[5]. In recent years, numerous techniques for automated counting systems employing image processing have been proposed in various fields. Fish egg count, on the other hand, is an area that has received comparatively little attention. Ramdhan and Syafrullah [12] proposed counting eggs based on some morphological operation performed over images of eggs.A simple method for automatically counting feeder fish using image processing techniques was described by Toh et al. [14] A video of a school of fish is captured, and each frame is processed separately and independently. The first step is to obtain blobs that represent the positions of the fish. Several methods for completing this task are discussed. Noise and background objects are removed from the blob image. The area information of the blobs is used to count the number of fish in one frame, and the average number of fish over all frames is then recorded.

In [7], Guo and Yu describe an efficient automatic cell counter.The counter makes use of histogram dual-threshold to split the cell and background and blob evaluation to come across the cell. The adherent cells are then segmented the use of the K-means clustering method.

In [1], Asmitaba and Pipalia used morphological operation, edge detection and segmentation to detect and count small objects. Wang et al. [15] presented a method to detects tiny objects in aerial images.Because tiny objects have a small number of pixels and are easily confused with the background, detecting them in aerial images remains a difficult problem. Because fish eggs are small, the problems are similar. Different segmentation techniques are described in [11], from which we got the idea about small object segmentation. They have discussed texture based, threshold based, super pixel based etc. image segmentation techniques.

The target of this research is to accurately segment and identification of fish eggs in images. This research is conducted over different fish eggs images. Primarily many algorithms can be considered to provide the desired result. There are many challenges in segmenting the fish eggs from the background. Such as

- Traditional clustering algorithms are very much dependent on the data in order to achieve a satisfying result. Those algorithms might work well for some particular datasets but not for all.

- Only morphological operation may not give the correct results for densely placed eggs.
- Only segmentation techniques are not able to give optimum results.
- In [5], Duan et al. developed an automatic counting system for pelagic fish eggs but the proposed system is semi-automatic. There have some limitations. They had developed a fixed setup for capturing images and the diameter of the dish is fixed.

Considering these drawbacks, we have proposed a method for counting fish eggs where the algorithm does not depend on dish sizes. Our proposed methods are discussed in the next chapter.

CHAPTER 3

METHODS AND METHODOLOGIES

Diverse methodologies and techniques, such as Bottom-Hat Transformation [13], differnet kinds of segmentation [17], and other kinds of thresholding techniques [10], have been tested under various circumstances in order to obtain proper counting of the number of eggs from photos of fish eggs placed on a plate [5]. A thorough proper analysis is required to fully comprehend the basic working principles and limitations of the algorithms, as well as state-of-the-art approaches for counting the number of fish eggs from an image.

3.1 Image Acquisition

Here we have used eggs of Rohu(*Labeo rohita*). Each sample of fish eggs is placed into a transparent dish with 10mm depth of water. We have dispersed eggs in the dish. We have captured the image from the top side of the dish using the smartphone camera. Here we were using Redmi Note 9 Pro smartphone, which has a 64MP camera. A total of 17 dish sample images have been captured.

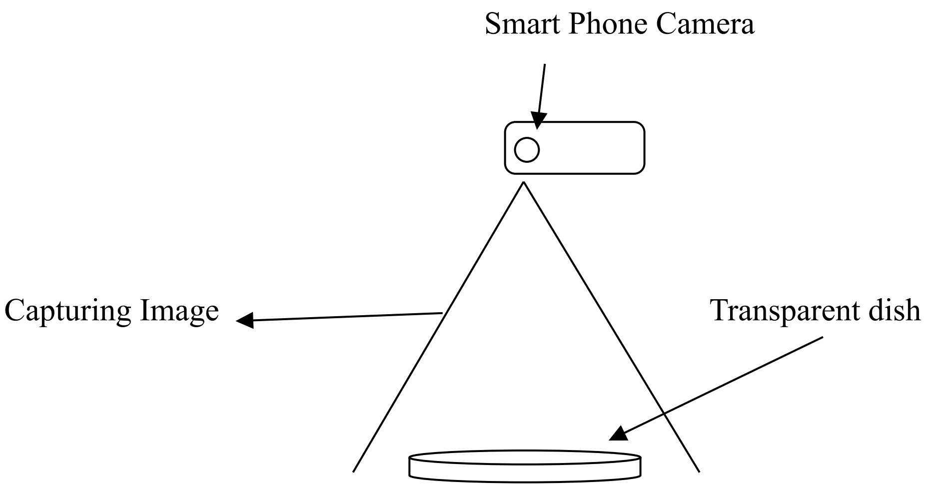
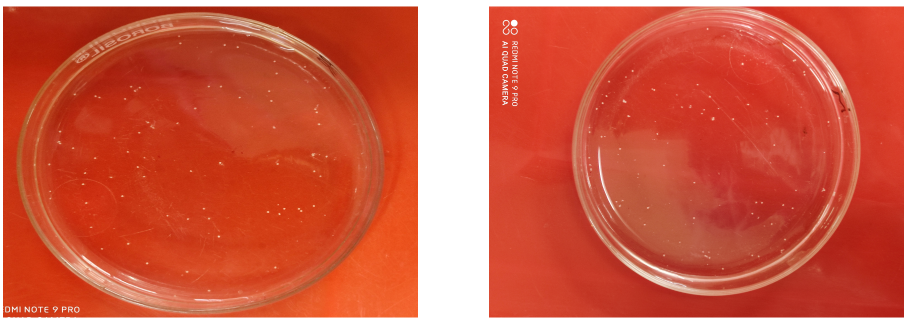


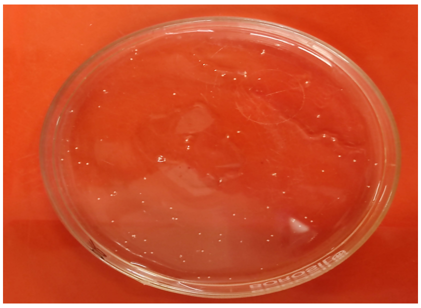
Figure 3.1: A mechanism of Fish Egg Image Capturing system

3.2 Dataset Description

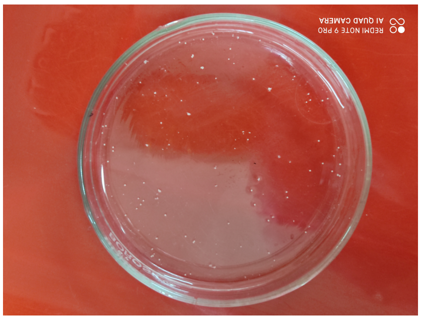
We have performed our experiment on the dataset, which has seventeen images of fish eggs placed on transparent dishes. The dataset has information about the number of eggs present on each data which is manually counted. The dataset is shown in the following Table 3.1.



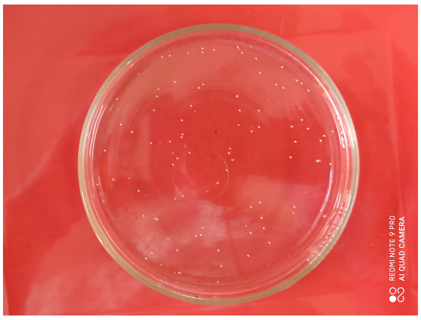
(a) (b)



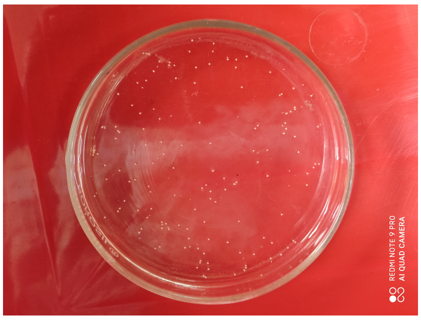
(c)



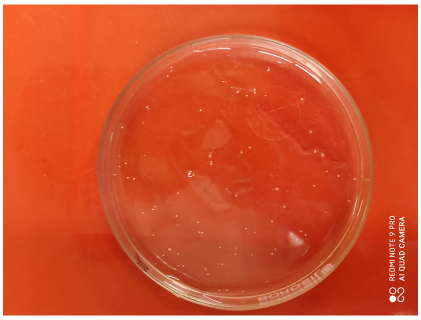
(d)



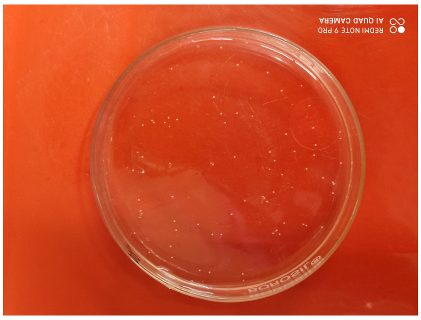
(e)



(f)



(g)



(h)

Figure 3.1: Sample images [(a)-(h)] of the Captured Image dataset

Sample Image#	Number of eggs using Manual Count
Fish#1	114
Fish#2	119
Fish#3	116
Fish#4	107
Fish#5	152
Fish#6	192
Fish#7	132
Fish#8	113
Fish#9	127
Fish#10	154
Fish#11	206
Fish#12	134
Fish#13	127
Fish#14	160
Fish#15	169
Fish#16	161
Fish#17	191

Table 3.1: Number of Fish Eggs Present in Each Image using Manual Counting

3.3 Image Pre-processing

As the positions of the dish locations in the images are not fixes; therefore, we must define a region of interest (ROI) based on the dish's edges from the images at the beginning. We have to ensures that any specular reflections on the dish's sides or from water droplets on the dish's base do not affect the final result. To avoid this kind of unwanted noises, we have performed some preprocessing operations before segmenting and counting eggs. Grey level morphological operations and image enhancement processing (such as Gamma Enhancement) have done on a grey scale representation of the ROI. The schematic of the preprocessing procedure shown in the following Figure 3.2.

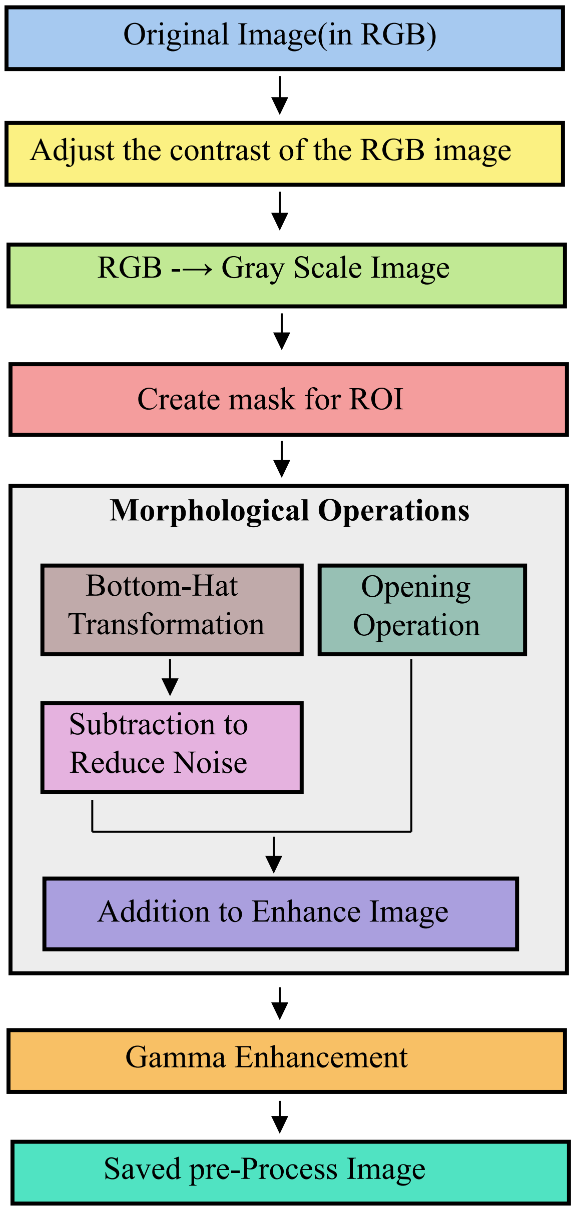


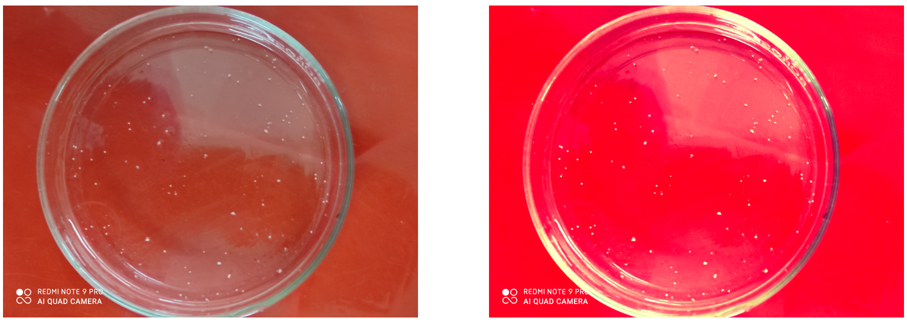
Figure 3.2: Image pre-processing flowchart

Algorithm 1 Image Pre-Processing

1: Read RGB input image I
2: Apply contrast adjustment on I
3: Create a mask for the detection of ROI
4: Convert RGB image(I) into grayscale image(G)
5: Perform Bottom-Hat transformation on G and save the output as G_1
6: Subtract G_1 from G to reduce noise and save the results in G_2
7: Do the opening operation on G and save the output in G_3
8: Add G_2 and G_3 and save the result in G_4
9: Perform Gamma correction on G_4
10: Save the resultant image after performing the Gamma correction

3.3.1 Contrast Adjustment of the Image

We have processed the fish egg image through several steps. First, we have taken the image and adjusted the contrast of the RGB image, specifying contrast limits to get better performance. Adjustment is made by experimenting several times and then fixing up the best values suitable for our dataset.



(a) Before Contrast Adjustment (b) After Contrast Adjustment

Figure 3.3: Contrast Adjustment of the Image

3.3.2 Convert RGB Image to Grayscale Image

After adjusting the image's contrast, we have converted the RGB image to grayscale for applying the following edge detection operation. Here we have used the Canny edge detector, which applies to a grayscale image. Red, green, and blue are the three colours of an RGB image. RGB channels use in computer displays and image scanners to nearly match the colour receptors in the human eye. But Grayscale images have one intensity channel. Generally, two methods are used to convert an RGB image into a grayscale image. The methods are -

- Average method
- Weighted method or luminosity method

Average Method

The average method is the most straightforward. It's as simple as taking the average of three colours. Because it's an RGB image, we need to calculate the average values of R, G and B on each pixel location to create a grayscale image. The intensity value of a grayscale image at pixel location (x, y) is calculated as follows:

$$I(x, y) = \frac{(R_{(x,y)} + G_{(x,y)} + B_{(x,y)})}{3}$$

where $R_{(x,y)}$, $G_{(x,y)}$ and $B_{(x,y)}$ are the intensity of Red,Green and Blue channels at pixel location (x, y) in the RGB image.

Weighted Method or Luminosity Method

This issue arises when we take the average of the three-channel intensities. Because the three distinct colours have three different wavelengths, each contributing to the creation of the image, we must take an average based on their contribution rather than applying the equal-weighted average approach. Suppose we use w_1, w_2, and w_3 weights for Red, green and blue channels, respectively, to calculate the intensity of a grayscale image. Now, the intensity value of the grayscale image at pixel location(x,y) is rewritten as follows:

$$I(x, y) = \frac{(w_1 \times R_{(x,y)} + w_2 \times G_{(x,y)} + w_3 \times B_{(x,y)})}{3}$$

(a) Contrast Adjusted Image in RGB (b) Grayscale Image

Figure 3.4: RGB Image to Grayscale Image Conversion

3.3.3 Create Mask for ROI

We have created a mask to understand our ROI better. Here we have used MatLab function *createMask()* to create a mask.

Canny Edge Detector

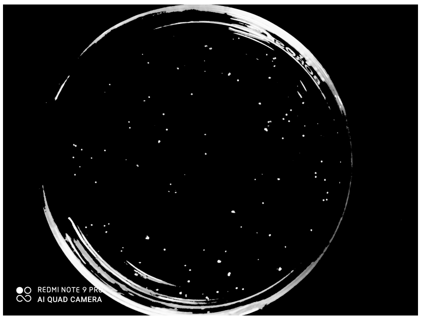
An edge is a collection of connected pixels that forms a boundary between two disjoint regions. Canny Edge Detection[4] is a popular edge detection algorithm.It was developed by John F. Canny in 1986. It is an optimal edge detector that takes a grayscale image as input and finds edges based on intensity discontinuities.[1] All

[1] https://towardsai.net/p/computer-vision/what-is-a-canny-edge-detection-algorithm

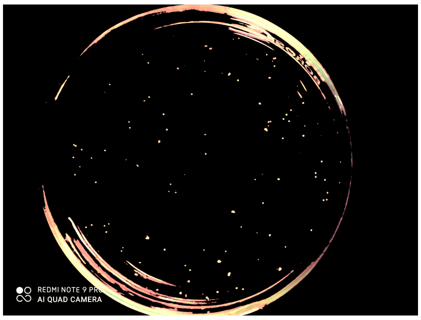
the steps are as follows: Algorithm 2.

Algorithm 2 Canny Edge Detection Algorithm

1: Apply Gaussian filter to smooth the image in order to remove the noise.
2: Find the intensity gradients of the image.
3: Apply Non-maximum suppression
4: Apply double threshold to determine potential edges.
5: Track edge by hysteresis: Finalize the detection of edges by suppressing all other edges that are weak and not connected to strong edges.



(a) Grayscale Image



(b) Mask Created from Grayscale Image

Figure 3.5: Mask Generated from Grayscale Image

3.3.4 Morphological Operations

Morphological Operations are a class of image processing operations that manipulate digital images based on their shapes. Each image pixel in a morphological operation corresponds to the value of another pixel in its neighbourhood. We can construct a morphological operation that is sensitive to specific shapes in the input image. It can be done by selecting the shape and size of the neighbourhood pixel. Morphological operations use a structuring element to create an output image of the same size as the input image.

Bottom-Hat Transformation

Bottom-hat filtering is the same as subtracting the input image from the output of a morphological closing operation on the input image.Bottom-Hat[8] transformation is used to generate a frame that represents the change in illumination across the image. Here we use *imbothat(I,nhood)* function in MatLab, where *'I'* is the input image which to be transformed using Bottom-Hat transformation and *'nhood'* is a matrix of 0s and 1s that specifies the structuring element neighbourhood. We use *'nhood'* value 10 for which we get a better result.

After performing the bottom-hat transformation, we have performed subtraction between the image, which was taken as input to the Bottom-Hat transformation and the output of the Bottom-Hat transformation to reduce the noise from the image.

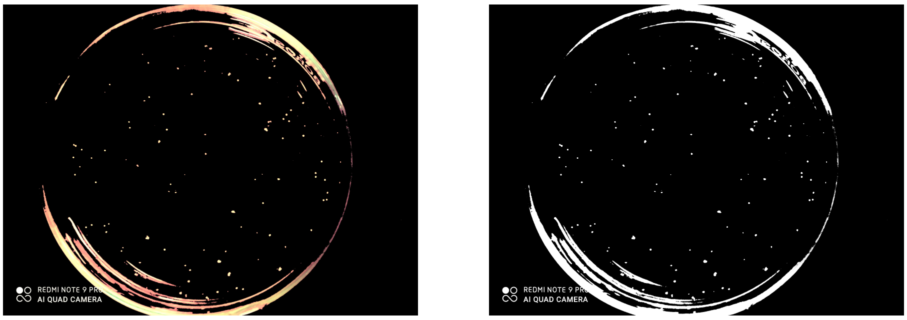
Opening Operation

Opening operation[9] is a morphological operation. It is used for removing small objects from images. The opening can be performed in two steps. First, perform erosion operation is performed on the image to shrink the object by removing boundary pixel. Then, apply the dilation operation to add a layer of pixels to the inner and outer boundaries.

Figure 3.6: Opening Process

Here we have used another copy of the image as input to the opening operation, which is input into the Bottom-Hat transformation. We use the disk as a structuring element for opening operation. In MatLab, we use *imopen(I,strel('disk',4))* function to perform the opening operation where *'I'* is the input image and *strel('disk',4)* function creates a structuring element of a disk which has a radius four unit.

We have added the output of the subtraction operation and the output of the opening operation to enhance image quality.



(a) Before Morphological Operations (b) After Morphological Operations

Figure 3.7: Effects after Performing Morphological Operations (Bottom-Hat Transformation, Opening Operation, Subtraction to Reduce Noise, Image Addition)

3.3.5 Gamma Enhancement

Gamma correction is used after performing linear operations such as addition, subtraction, and multiplication on an image. It is a nonlinear adjustment which is used to maintain the stable gamma value. If the gamma value is too large or too small, it can result in poor contrast.

Here we have performed pixel-wise subtraction and addition operations, so we need to correct the gamma value to get a better image with a stable gamma value. All preprocessing operations are done over the image. Now we save our preprocessed image for further process.

3.4 Segmentation of the Eggs

We have taken the saved preprocessed image as input and performed various segmentation and morphological operation for segmenting the eggs from the background. Step-by-step processes are shown in the Figure 3.8.

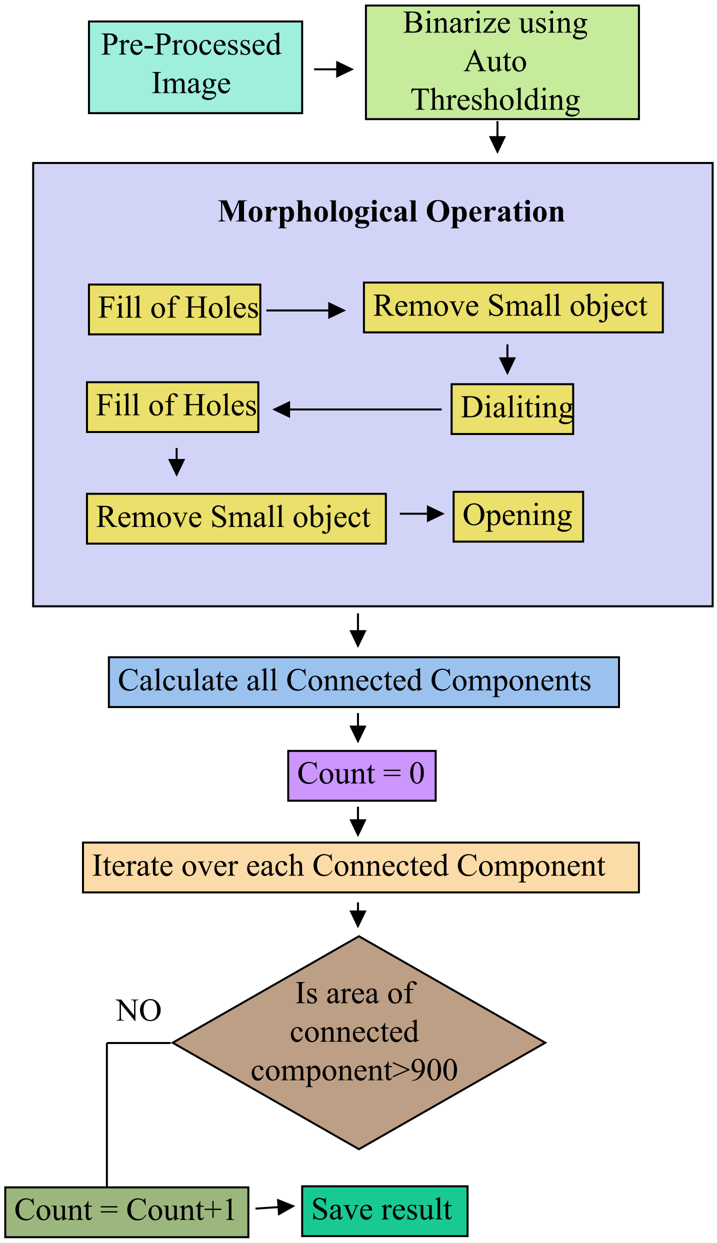


Figure 3.8: Image segmentation flowchart

Algorithm 3 Automated Egg Counting

Read Pre-Processed Image I from disk.
Binarize the image I using auto thresholding and saved in B
Do the fill holes operation on B and save the result in B_1
Do the open operation on B_1 and save the result in B_2
Perform Dilation on B_2 and save result in B_3
Again perform fill holes followed by an opening operation on B_3 and save result in B_4
Find all connected components CC present inside $B4$
$eggCount = 0$
for cc in CC **do**
  /* area(cc) is a function which returns the area of component cc based number of pixels present inside cc */
  if area(cc)<900 **then**
    $eggCount += 1$
  end if
end for
Return eggCount

3.4.1 Binarization of the Image

Even after preprocessing, the image mean grey level intensity varied between images due to several uncontrollable factors (different location and number of eggs, previous and unequal light exposure, etc.). To compensate for this, the images are binarized using adaptive thresholding [10] into pixels representing eggs and pixels representing background.

Algorithm 4 Otsu's Automatic Thresholding Method

1: For an image I on X gray levels, compute the gray-level histogram $h(0), h(1), ..., h(X-1)$. Normalize the histogram by dividing through by the number of pixels in I—the histogram now represents the probability of each gray-level.

2: For each possible threshold $t = 0, ..., X-2$ partition the histogram into background B (gray-levels less than or equal to t), and foreground F (gray-levels more than t).

3: Compute $\sigma_{B(t)}, \sigma_{F(t)}$, the variance of the background and foreground gray-levels. Compute the probability of a pixel being background

$$\mathcal{P}_B(t) = \sum_{j=0}^{t} h(j)$$

and $\mathcal{P}_F(t)$ similarly.Set

$$\sigma(t) = \mathcal{P}_B(t)\sigma_{B(t)} + \mathcal{P}_F(t)\sigma_{F(t)}$$

and select as threshold $t_{min} = min(\sigma(t))$.

After getting the optimal threshold value (t_{min}), we have used the basic thresholding segmentation algorithm to binarize the image.

Algorithm 5 Basic Threshold based Segmentation

Input: f is the input image, t_{min} is threshold value calculated using Otsu's method

Output: g is an binary image

1: Search all pixel $f(x, y)$ in input image f
2: **if** $f(x, y) \geq t_{min}$ **then**
3: $\quad g(x, y) = 1$
4: **else**
5: $\quad g(x, y) = 0$
6: **end if**

The above algorithm gives an output of a binary image.

3.4.2 Morphological Operations

After binarization, there have been some problems with fish eggs, such as egg objects may have holes, some small noises may still exist, some parts of the fish eggs may be cut out, and some eggs may be touched by one another. We have performed some morphological operations to get better results. First, we have performed fill holes operations followed by an open operation which removes small objects from the binary image. We have specified the value of the size filter is 90 in MatLab function *bwareaopen(h1, size)*. All objects less than this size have been considered noise and removed. After this, we performed dilatation to grow and thicken objects so that divided parts of eggs were connected. Following that, the procedures for filling holes and removing small objects were repeated. Finally, an opening operation was used to remove, break, and diminish false connections between egg objects.

3.4.3 Connected Components

Scan an image and form groups of pixels known as connected components based on pixel connectivity, i.e. all pixels in a connected component have similar pixel intensity values, and there must exist connectivity. Here we have used the connected components to calculate the no. of eggs. We have calculated the area of each connected component. If the area of the connected component is greater than 900, then it is considered as an egg. Connected component labelling for binary image procedure is shown in Algorithm 6.

Scan the input image from left to right and top to bottom. Let P be a pixel at any step in the scanning process.

R is the pixel which is right adjacent to P & T is the pixel which is top adjacent to P.

Before P is scanned, R and T are already scanned.Assumed, we use 4-connectivity in the Algorithm 6.

I(P) denotes intensity value of pixel P.

L(P) denotes label assigned to pixel P.

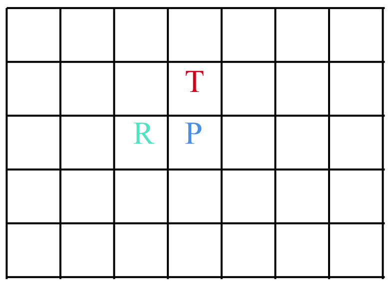
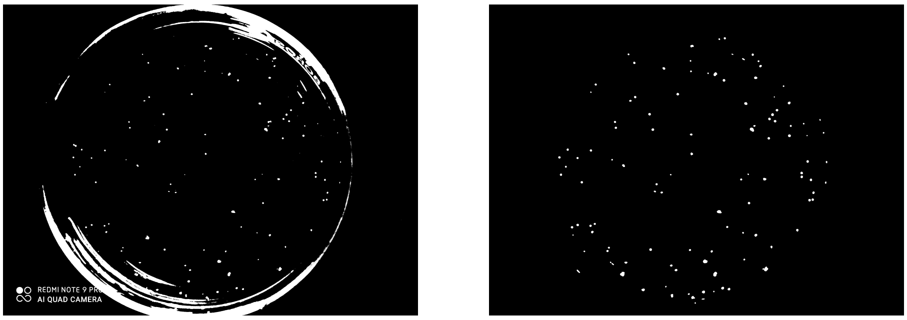


Figure 3.9: Position of Pixel P

Algorithm 6 Connected Components labelling

Scan the input image from left to right and top to bottom. Let P be a pixel at any step in scanning process.
if I(P) = 0 **then**
    move to next scanning position
end if
if I(P) = 1 and I(R) = I(T) = 0 **then**
    assign a new label to position P
end if
if I(P) = 1 and one of two neighbor is 1 **then**
    assign its label to P.
end if
if I(P) = 1 both I(R) = I(T) = 1 **then**
    if L(R) = L(T) **then**
        L(P) = L(R)
    end if
    if L(R) != L(T) **then**
        assign one of the labels to P and make a note that two labels are equivalent.
    end if
end if
One more time scan the image from beginning replace each label by the label assigned to its equivalence class



(a) Pre-processed Image (b) after Segmentation

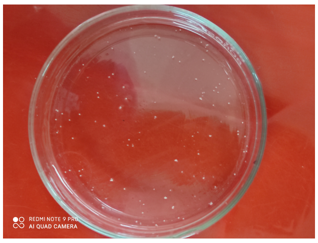
Figure 3.10: Effects after segmentation

CHAPTER 4

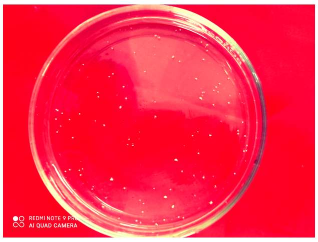
RESULTS AND DISCUSSION

In this section we have discussed about the performance of our automated system. Here, we have compared our results with respect to manual count as well as time required to perform the both type of operations. Our automated system has taken far less time compared to the manual process. As discussed, we have sent our images through the pre-processing procedure followed by the segmentation and counting process. We have reported our results in Table 4.1 to compare with the manual count.

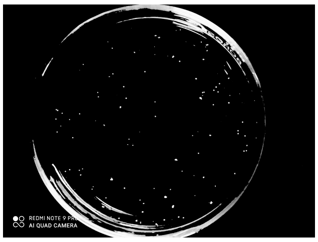
We have noticed each step of our procedure what's happening with our image. We have saved images on each step. Changes on an image in each step are shown in Figures 4.1 and 4.2



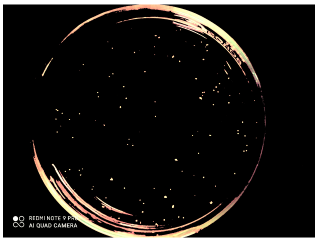
(a) input image



(b) after contrast adjustment

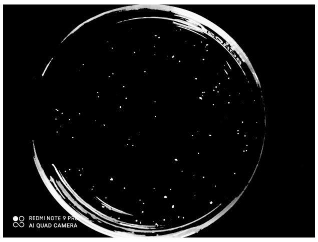


(c) converted to grayscale

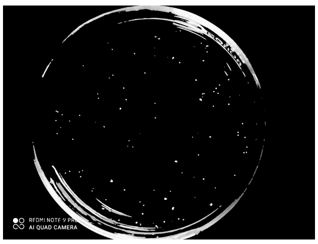


(d) mask created

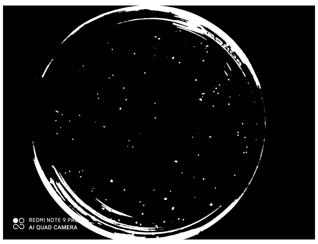
(e) after Bottom-Hat transformation



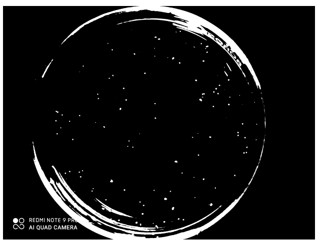
(f) after subtraction



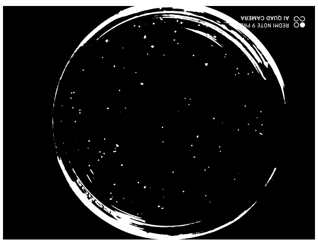
(g) after opening



(h) after addition

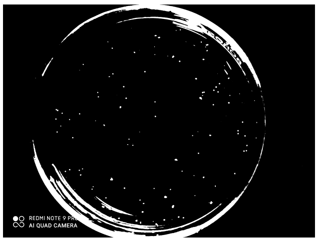


(i) after gamma enhancement

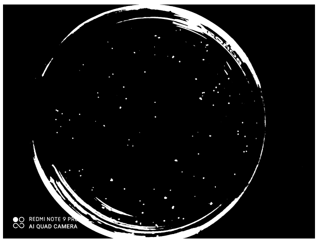


(j) Pre-processed image

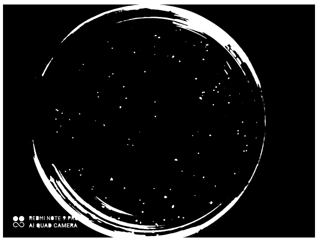
Figure 4.1: Images produced in each step of Pre-processing



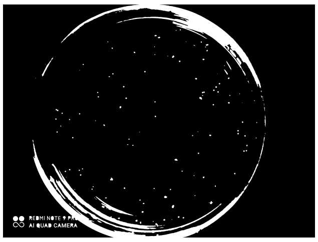
(a) pre-processed image



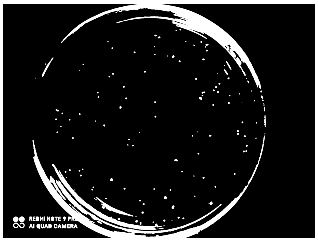
(b) after binarization



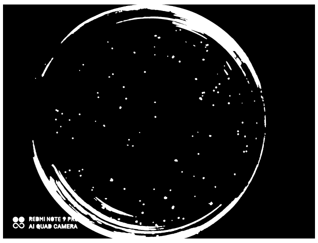
(c) after fill of holes



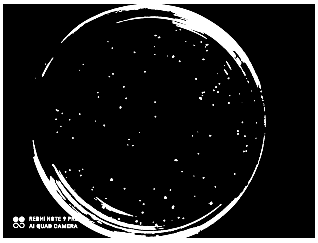
(d) after removing small object



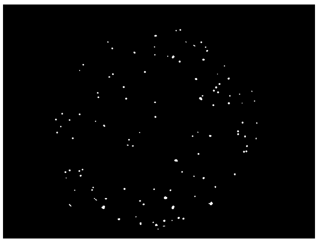
(e) image after dilation



(f) image after removing small object



(g) image after opening



(h) get eggs only using connected component

Figure 4.2: Segmentation of Eggs from Background

4.1 Manual vs Automatic Counts

We have run an experiment over seventeen fish egg images. During the experiment, we recorded our results in Table 4.1 to compare our automated output with the manual count.

Sample Image#	Number of eggs using Manual Count	Number of eggs using Automatic Count
Fish#1	114	116
Fish#2	119	121
Fish#3	116	117
Fish#4	107	109
Fish#5	152	155
Fish#6	192	189
Fish#7	132	133
Fish#8	113	110
Fish#9	127	127
Fish#10	154	151
Fish#11	206	209
Fish#12	134	135
Fish#13	127	129
Fish#14	160	161
Fish#15	169	172
Fish#16	161	159
Fish#17	191	193

Table 4.1: Number of eggs using Manual and Automatic counting

Based on Table 4.1 data we have calculated the overall error by using the following formula.

$$E = \frac{1}{n}\sum_{i=0}^{n}(y - \hat{y})^2$$

Where,

E = average error in automatic counting with respect to manual counting

n = number of images

$\hat{y}$ = automatic counting value

y = manual counting value

We have found overall error is 4.823529412, which is not comparatively large.

We correlated the numbers estimated using image analysis with manually counted eggs in 17 samples for algorithm calibration and verification. We found a correlation(R^2) value is 0.995. The high correlation between the two methods indicated that the automatic counts is accurate.

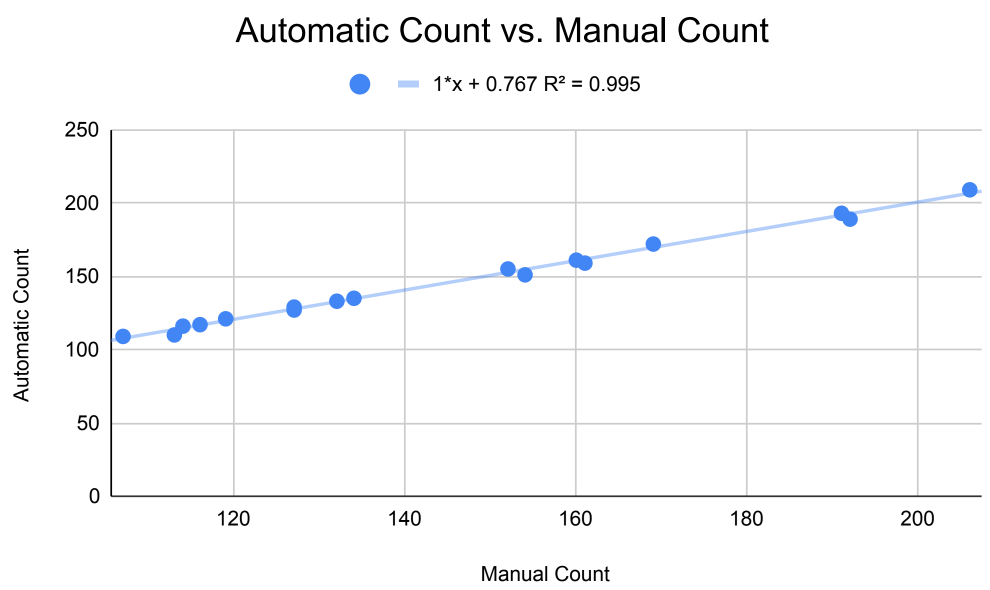


Figure 4.3: Comparison between manually counted egg numbers and numbers counted using the proposed method

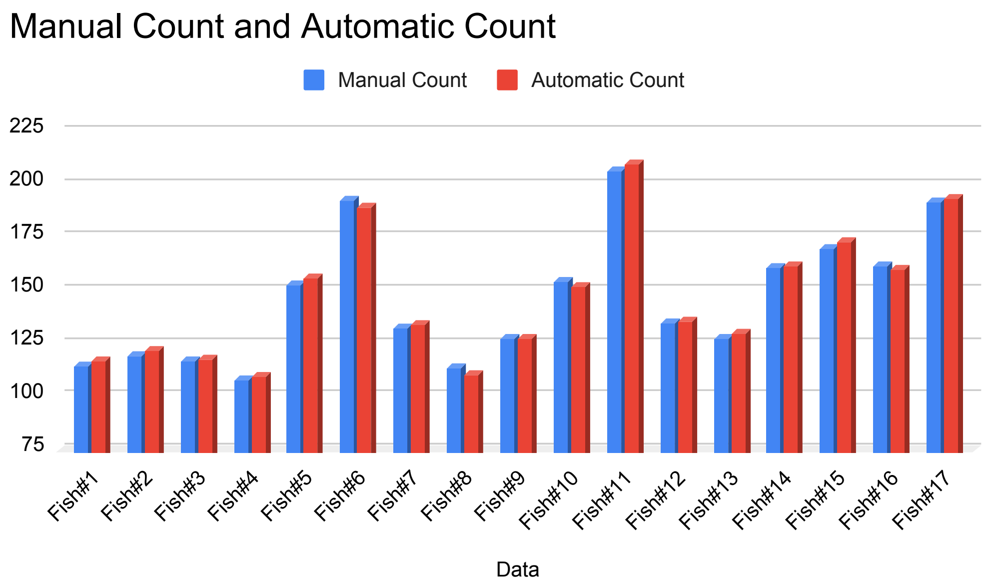


Figure 4.4: Automatic vs Manual Counting of Fish Eggs

We have recorded the time taken for the manual counting. Table 4.2 below shows the information about the duration of manual counting of the fish eggs for 17 images each. The average duration has been calculated by using the Eq. 4.1 below:

$$t_{avg} = \frac{1}{n} \sum_{i=0}^{n} t_i \tag{4.1}$$

where t_i and n represent the counting time for i^{th} image and number of total images, respectively.

Sample Image#	Number of eggs using Manual Count	Counting Time(sec)
Fish#1	114	137
Fish#2	119	145
Fish#3	116	139
Fish#4	107	121
Fish#5	152	163
Fish#6	192	197
Fish#7	132	156
Fish#8	113	131
Fish#9	127	147
Fish#10	154	203
Fish#11	206	194
Fish#12	134	160
Fish#13	127	134
Fish#14	160	150
Fish#15	169	177
Fish#16	161	183
Fish#17	191	203

Table 4.2: Duration of Counting Eggs Manually

From Table 4.2 we have calculated the average time taken to count the fish eggs manually is 161.1764 ± 26.86129952 sec.

We have recorded the time taken for the automated counting. Table 4.3 below shows the information about the duration of automated counting of the fish eggs for 17 images each.

Sample Image#	Number of eggs using Automated system	Counting Time(sec)
Fish#1	116	9
Fish#2	121	8
Fish#3	117	10
Fish#4	109	9
Fish#5	155	13
Fish#6	189	15
Fish#7	133	9
Fish#8	110	10
Fish#9	127	11
Fish#10	151	12
Fish#11	209	16
Fish#12	135	2
Fish#13	129	6
Fish#14	161	10
Fish#15	172	9
Fish#16	159	7
Fish#17	193	13

Table 4.3: Duration to Counting Eggs Using Automated System

We have used above Eq. 4.1 to calculate average time. From Table 4.3 we have calculated the average time taken to count the fish eggs using automated system is 9.9411 ± 3.362859428 sec.

Based on Tables 4.2 and 4.3, the time to process the image for the automatic counting process is 16 times faster compared to the manual counting process.

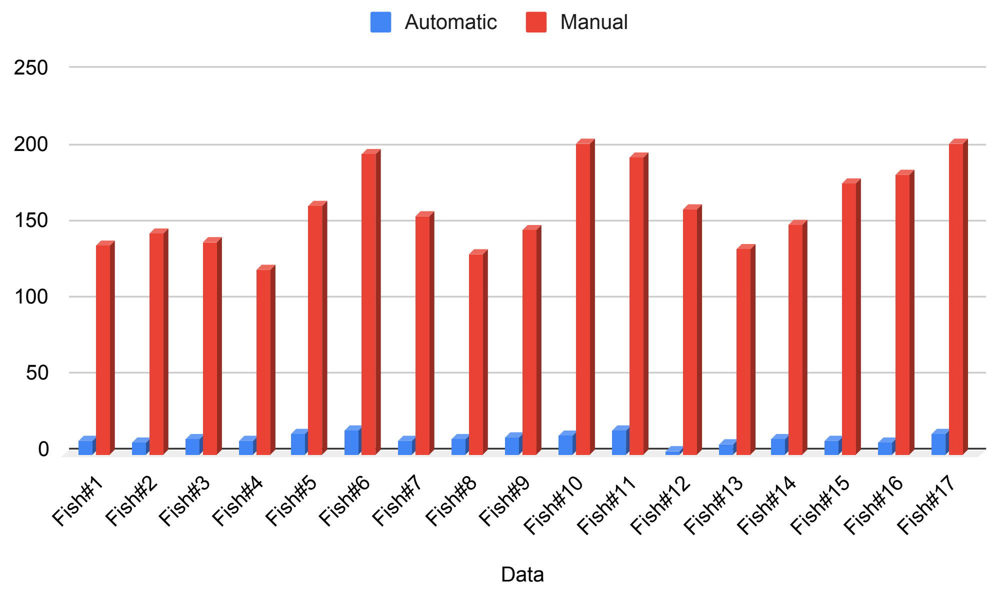


Figure 4.5: Comparison between time(sec) taken with automatic and manual counting

4.2 Design and Development of AFECS

This section shows how AFECS (Automated Fish Egg Counting System) works, from uploading an image of a fish egg to counting the number of fish eggs based on the uploaded images. The design of the system doesn't have any security features (like sign in or register) because anyone can use it. In future updates, the security features could be improved. When the user opens the system, it will take them to the main page, where they can upload an image of fish eggs on a dish. Figure 4.6 shows that this page has several functions, such as the "Upload" and "Count" buttons. The counting process is the last step in the automated system for counting fish eggs. The counting is done using our method, which was already explained.

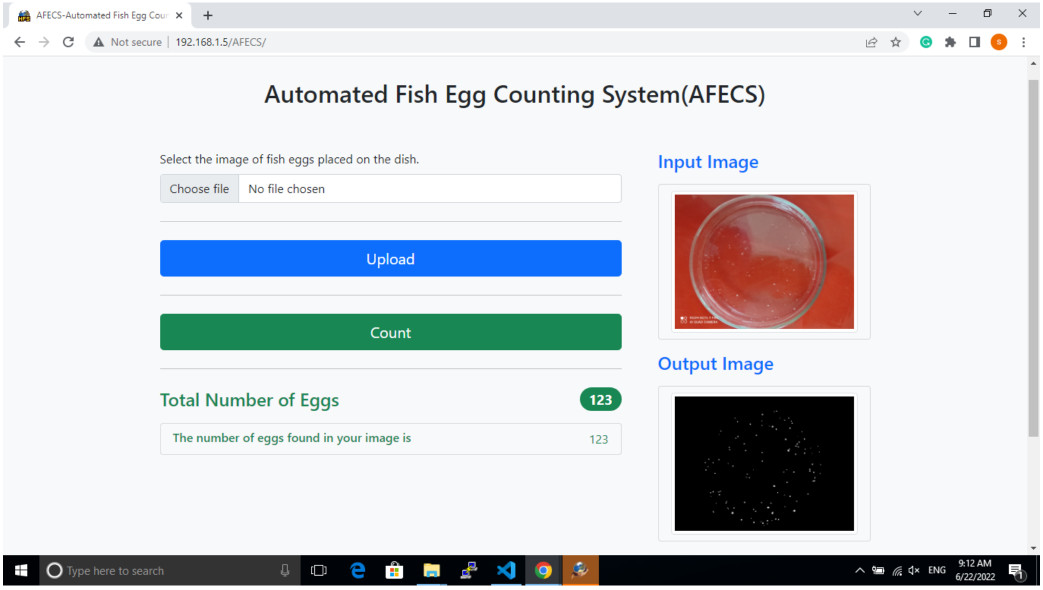


Figure 4.6: A glimpse of the developed AFECS

The user can count the result by clicking the “Count” button after the image has been uploaded by clicking the “Upload” button. A binary image will be produced where eggs are segmented from the background, as well as the total number of fish eggs that appear, as shown in Figure 4.6.

4.3 Experimental Analysis and Discussion

The processing time for automatic egg counting will vary depending on the computer’s speed. Each image took about 33 seconds to process using our system. This processing did not take long and thus did not require many man-hours. When there are about 100 eggs on a dish to count, it takes around 3 minutes or more. When there are more eggs, manually counting them becomes extremely difficult. Furthermore, the results may contain some errors. When compared to manual methods, the results of our experiments show that our algorithms are more reliable, flexible, non-destructive, and fast. Our experiment, however, shows that the system has some limitations. To begin, while removing specular reflections at the dish's edge, some eggs located in the reflections will be difficult to detect. Second, the original RGB contrast is a little low due to the high transparency of fish eggs.These factors all increased the difficulty of image processing and also affected the accuracy of identification and counting. To improve the method we will in the future try to optimize image acquirement by using a digital camera of higher the resolution, and finding a good method to avoid eggs close to the edge of the dish.

CHAPTER 5

CONCLUSION

We have created an automatic counting method for fish eggs using the combination of Otsu's threshold method, morphological operations, and other image processing techniques. The method has applied for counting the Rohu fish eggs. We believe the method could be applied to a variety of other fish eggs. Automatic and manual counting has been compared to validate the method's accuracy. The automatic eggs counting algorithm has a very lower average counting error which is well within acceptable levels for many types of work. The main advantage of the automated counting system over traditional manual counting methods is that it reduces the time and cost of the human. The proposed methodology can be applied on other data samples of different fish species which would be explored in future.

REFERENCES

[1] Gohil Asmitaba and Dhaval S. Pipalia. Design an algorithm to detect and count small size object using digital image processing. 2016.

[2] Bernal, Miguel and Somarakis, S. and Witthames, and van Damme, Cindy and Uriarte, Andrés and Lo, Nancy and Dickey-Collas Mark. Egg production methods in marine fisheries; an introduction. *Fisheries Research*, 117:1–5, 04 2012.

[3] H. C. Boyar and R. A. Clifford. An automatic device for counting dry fish eggs. *Transactions of the American Fisheries Society*, 96(3):361–363, 1967.

[4] John Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–698, 1986.

[5] Yane Duan, Lars Helge Stien, Anders Thorsen, Ørjan Karlsen, Nina Sandlund, Daoliang Li, Zetian Fu, and Sonnich Meier. An automatic counting system for transparent pelagic fish eggs based on computer vision. *Aquacultural Engineering*, 67:8–13, 2015.

[6] The Financialexpress. India's blue economy net getting bigger! country ranks third in fisheries and second in aquaculture. `https://www.financialexpress.com/opinion/indias-blue-economy-net-getting-bigger-country-ranks-third-in-fisheries-and-second-in-aquaculture/1867607/`, May 2022.

[7] Xiaomin Guo and Feihong Yu. A method of automatic cell counting based on microscopic image. volume 1, pages 293–296, 08 2013.

[8] Rafsanjany Kushol, Md Sirajus Salekin, A. B. M. Ashikur Rahman, and Md Kabir. Contrast enhancement by top-hat and bottom-hat transform with optimal structuring element: Application to retinal vessel segmentation. pages 533–540, 07 2017.

[9] Khairul Anuar Mat Said, Asral Jambek, and Nasri Sulaiman. A study of image processing using morphological opening and closing processes. *International Journal of Control Theory and Applications*, 9:15–21, 01 2016.

[10] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66, Jan 1979.

[11] Km Pooja and Reghunadhan Rajesh. Image segmentation: A survey. pages 521–527, 08 2016.

[12] Syainul Ramdhan and Muhammad Syafrullah. Fish eggs calculation models using morphological operation. In *2019 6th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*, pages 392–397, 2019.

[13] Suman Thapar and Shevani Garg. Study and implementation of various morphology based image contrast enhancement techniques. *Int. J. Comput. Bus. Res*, 128:2229–6166, 2012.

[14] Y. Toh, T. Ng, and Beng Liew. Automated fish counting using image processing. *International Conference on Computational Intelligence and Software Engineering*, 12 2009.

[15] Jinwang Wang, Wen Yang, Haowen Guo, Ruixiang Zhang, and Gui-Song Xia. Tiny object detection in aerial images. pages 3791–3798, 01 2021.

[16] P. R. Witthames and M. Greer Walker. An automated method for counting and sizing fish eggs. *Journal of Fish Biology*, 30(3):225–235, 1987.

[17] Nida M. Zaitoun and Musbah J. Aqel. Survey on image segmentation techniques. *Procedia Computer Science*, 65:797–806, 2015. International Conference on Communications, management, and Information technology (ICCMIT'2015).