**SNAKE SCALES COUNTING AND SPECIES RECOGNITION BASED ON CONTOUR COUNTING**

**A report on**

**Computer Vision Lab Project**

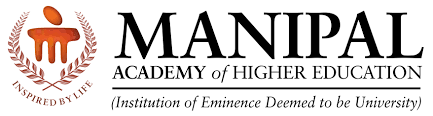
**[CSE-3181]**

Submitted By

**Sreya Chunduri, 210962206**

**Anoushka Nag, 210962096**

**Saksham Arora, 210962202**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**MANIPAL INSTITUTE OF TECHNOLOGY,**

**MANIPAL ACADEMY OF HIGHER EDUCATION**

**NOVEMBER, 2023**

**SNAKE SCALES COUNTING AND SPECIES RECOGNITION BASED ON CONTOUR COUNTING**

Sreya Chunduri, Anoushka Nag, Saksham Arora

Manipal Institute of Technology, India

Manipal Institute of Technology, India

Manipal Institute of Technology, India

[28sreyachunduri@gmail.com](mailto:28sreyachunduri@gmail.com) ; [anoushkanag03@gmail.com](mailto:anoushkanag03@gmail.com) ; [saksham.study19@gmail.com](mailto:saksham.study19@gmail.com)

***Abstract – In this project, we aim to identify and count the number of scales on a snake’s body using contour counting. Before contour counting, the image undergoes certain preprocessing techniques to get a sharper, noise-free image. Count of the number of snake scales can be used to classify whether the snake is venomous or non-venomous. This has critical real-life use cases in research work and especially when it comes to the safety of human race against deadly snake bites.***

***Keywords – Snake, Scale counting, Computer Vision, OpenCV, Contour Counting, Scales***

1. **Introduction**

Accurate identification and counting of snake scales play a crucial role in ensuring human safety, particularly in the context of deadly snake bites. Worldwide, there are around 3,000 snake species, and an alarming 20% of them are venomous, with 7% posing a direct threat to humans. In a country like India, snakebites result in a tragic toll of 50,000 lives lost each year. This emphasizes the pressing need for research that can provide innovative solutions to this life-threatening issue.

This paper is dedicated to contributing to this essential research effort by presenting a new method for counting snake scales, relying on contour counting and image processing. Accurate scale counting is instrumental in species identification, a critical factor in administering antivenom promptly, thus reducing fatalities. It can cater to the needs of both researchers and healthcare providers dealing with snakebite cases.

In the subsequent sections, we will delve into the methodology, benefits, and potential applications of our snake scale counting system, underlining its potential to have a significant impact on mitigating the risks associated with venomous snake encounters and, ultimately, safeguarding human lives.

1. **Literature review**
2. *Automated fish counting using image processing:*

In this research, the authors developed an automated fish population counting method for farmed Atlantic salmon in aquaculture using machine vision and a hybrid deep neural network model. Traditional manual fish counting methods are laborious and disruptive to fish. They proposed a solution by creating a hybrid neural network model, combining multi-column convolutional neural networks (CNNs) with dilated CNNs . The results demonstrated that the proposed model achieved a remarkable counting accuracy of 95.06%, with a high Pearson correlation coefficient of 0.99 compared to ground truth data. This approach outperformed CNNs and multi-column CNNs. Fish shoal dispersal can be better understood by looking at the density maps that the model produces; these maps may be used for monitoring and managing aquaculture.

1. *Egg Counting System Using Image Processing and a Website for Monitoring:*

The paper presents a novel method for real-time egg counting using image processing and IoT technology. The paper addresses the labour-intensive process of manually counting eggs on conveyor belts in the egg industry, emphasizing the need for a more efficient solution. The method involves using a Raspberry Pi camera to capture and process egg images. Image processing includes converting images to grayscale, applying a Gaussian filter, and creating a binary image. An opening operation is used to remove small objects, and eggs are counted using a contoured matrix.

1. *Automated Viral Plaque Counting Using Image Segmentation and Morphological Analysis:*

In the publication by Michael Moorman and Aijuan Donga a technique for automating the counting of viral plaques in photos is explained. The authors have separated viral plaques from background noise using image segmentation techniques. After that, morphological analysis has been used to identify traits and count the plaques. The study's method for counting plaques shows a high level of efficiency and precision, which lessens the need for manual labour in virology research. This method may save testing time and enhance the repeatability of viral plaque assay results. The experimental data are presented, the technique is discussed, and the ramifications for virology research are explored in this work. It provides a useful tool for streamlining a labour-intensive procedure and raising the output of research.

1. *A Web Platform for Measuring Frog Size and Counting via Contour Detection:*

The paper describes the creation of a web platform that uses contour detection and image segmentation techniques to measure frog size and count frogs in photos. The platform's goal is to automate ecological research studies on frog population monitoring. It makes assessments of the frog size by using computer vision algorithms to identify and define frog features in the photos. The system also counts the number of frogs in an image, which makes it useful for population assessments. Because the platform is web-based, a variety of researchers can collect data more easily and conveniently because it is easily available. With the use of this technology, frog population monitoring could become more accurate and efficient, supporting efforts to conserve the environment. The approach is explained in the publication, along with the experimental findings.

1. *Research on Steel Bar Detection and Counting Method based on Contours:*

The research study by Liu Xiaohu and Ouyang Jineng is based on the detection and counting of steel bars using contour-based approaches. The research, which is connected to the Naval University of Engineering's College of Electrical Engineering in Wuhan, China, mentions the difficulties in automating the recognition and counting of steel bars in pictures. The authors have provided a method for that to produce correct outcomes which depends on contour detection techniques. Their process divides apart the shapes of steel bars in photos so that counting may be done quickly. In addition to providing prospective solutions for expediting steel bar detection and counting procedures, the project emphasizes the application of computer vision in the engineering and construction arena. This activity can reduce manual labor and improve accuracy, which is beneficial for infrastructure and building projects.

1. *An Overview of Contour Detection Applications:*

In this an overview of contour detection methods is provided. It is connected to the Chinese Academy of Sciences' Institute of Automation's Research Center of Precision Sensing and Control. Their study examines numerous approaches and strategies for contour detection in images using computer vision techniques and image processing. It offers insights into how contour detection techniques have developed and how they are used in fields like scene analysis and object recognition. The writers have examined both conventional and deep learning-based methods, mentioning the benefits and drawbacks of each. For scholars and practitioners interested in contour detection in computer vision, this publication provides a comprehensive resource.

1. **methodology**

A diagram of a software flowchart

Description automatically generated

Fig 1. ALGORITHM

1. Image Preprocessing

The images in the given dataset are in a lab setup. Existing overhead lighting There are elements present in all the images that do not pertain to the snake body and patches of extreme brightness and dull spots. Hence removing unwanted elements and normalisation is essential. For the purpose of this paper, we have adopted two methods:

* 1. Manual Selection of Edge Points: The image is first converted to grayscale. Truncated thresholding is applied to eliminate extreme values beyond a certain threshold, outperforming other thresholding methods by preserving essential data features while reducing the impact of outliers or noise in the dataset. Image sharpening is performed using adaptive histogram equalisation that results in a far more enhanced and balanced image. We enable the user to manually click on the edge points encompassing the snake body. On applying the warped perspective transform, further image processing techniques described in Section B are followed.
  2. Image Segmentation: To perform segmentation, the approach is to subtract the background and only retain the foreground. We first convert the image to a grayscale image for processing. We detect contours in the binary image to compress horizontal, diagonal and vertical segments, keeping only the end points. The largest contour is selected and is drawn on an empty mask of zeros. This creates a black background for the object. Using bitwise and the initial image and the mask are combined resulting in a well segmented image showing only the snake’s body with scales against a black background. Following this truncated thresholding and image sharpening is performed, as in the manual selection method explained in Methodology Section A(i).

1. Canny Edge Detection

Canny Edge Detection, a standard edge detection algorithm has been used to detect edges. However, instead of utilising OpenCV’s inbuilt canny function, we have designed our own Canny edge detector which seems to outperform the inbuilt function for our cause. The difference occurs due to the custom thresholding where we are at liberty to test out different upper and lower threshold values according to the purpose. The inbuilt function relies on fixed, automatic thresholding which is not optimal for all images. With custom implementation, the noise removal and blurring process is in our hands, with the kernel size being determined by us for our application. Even with the Hysteresis Thresholding step, the custom implementation allows us to fine-tune parameters to our project’s needs. As the images we have are quite noisy with unnecessary information and caried lighting, this is an exceedingly important step.

1. Morphological Operations

Morphological operations are important for enhancing the quality of edges in the image and for improving the accuracy of contour detection. These operations are crucial for preparing the image after canny edge detection for contour detection and counting.

1. Morphological Dilation: The boundaries of the foreground object are expanded by adding pixels around it, making It larger. It closes any small gaps and holes within the object, making it more solid and making the features thicker. In our case, we see that dilation sometimes causes overlap of features due to thickening of lines.
2. Morphological Erosion: Any noise and unwanted details in the image by shrinking the boundaries of the foreground object, making it smaller. It is useful in our case, to distinguish boundaries of objects in the image, as it erodes away the outer pixels, leaving boundary pixels. The thinning effect it has proves particularly useful in the contour detection step.
3. Contour Finding and Counting

For this step, we first take the complement of the morphed and thresholded image. The contours of this inverted image are found and their edges retained. On iterating through the list of contours detected, the contour area of each contour is found. Using this, the average area is calculated. If the area of the ith contour is greater than the average area \* scale factor ‘k1’ and less than the average area \* scale factor ‘k2’, the scale counter is incremented. This allows for slight deviations and room for error in contour area upto factors k1 and k2.

Finally, the contours found are drawn onto the original image without the mask and the contour count is displayed.

TABLE I  
 DIFFERENT STAGES OF METHOD 1 FOR RAT SNAKE: MANUAL SELECTION OF EDGE POINTS

|  |  |  |
| --- | --- | --- |
| Original Colour | A close-up of a computer mouse  Description automatically generatedGray | A close up of a snake skin  Description automatically generatedTruncated Thresholding |
| Close-up of a snake skin  Description automatically generated  Image Sharpening | A close-up of a grey surface  Description automatically generatedCropped and Warped Image | A black and white image of a black surface  Description automatically generatedCanny Edge Detection |
| Morphological Operations | A black and white image of a black and white pattern  Description automatically generatedThresholding | Output Image |

TABLE II

DIFFERENT STAGES OF METHOD 1 FOR TRINKET SNAKE: MANUAL SELECTION OF EDGE POINTS

|  |  |  |
| --- | --- | --- |
| A snake skin in a plastic container  Description automatically generatedOriginal Colour | A fish in a glass container  Description automatically generatedGray | A snake with a long tail  Description automatically generated with medium confidenceTruncated Thresholding |
| A close-up of a snake  Description automatically generated  Image Sharpening | A close-up of a fish scale  Description automatically generatedCropped and Warped Image | A black and white image of a snake  Description automatically generatedCanny Edge Detection |
| A black and white image of a black background  Description automatically generatedMorphological Operations | A black and white speckled background  Description automatically generatedThresholding | A close-up of a fish  Description automatically generated  Output Image |

Parameters involved in method 1:

*A. Threshold values for Canny*

Fixed parameters:

* 1. Kernel for Dilation and Kernel for Erosion: (9,9)
  2. Contour Area: area>avg\_area\*0.3 and area<avg\_area\*8

TABLE I (a)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Lower Threshold** | **Upper Threshold** | **Canny Image** | **Final Image** | **Type of Snake** |
| 20 | 255 | A black and white image of a black surface  Description automatically generated | Scale Count = 88(Most are not scales) | Rat |
| 74 | 255 | A black background with white squares  Description automatically generated with medium confidence | Scale Count = 5(Scales are not detected) | Rat |
| 34 | 205 | A black and white image of a black surface  Description automatically generated | Scale Count = 50 | Rat |
| 34 | 170 | A black and white image of a black surface  Description automatically generated | Scale Count = 57 - BEST | Rat |
| 80 | 255 |  | Scale Count = 27(Scales not detected) | Trinket |
| 40 | 255 |  | Scale Count = 156 | Trinket |
| 20 | 205 |  | Scale Count = 310(Most are not scales) | Trinket |
| 40 | 197 |  | Scale Count = 152 - BEST | Trinket |

As is visible in the Table 1(a), different values of upper and lower threshold affect the scale count drastically. In the first case, although the count shows 88 (highest count), the image is overpopulated with minuscule contours that do not account for actual scales. This is because the lower threshold is so low that it detects the dark ridges between scales as contours. Then the value dips very suddenly to 5 as the lower threshold is so high that even slightly darkened pixels that are scales, are not coming forth. The count rises after this point when both upper and lower thresholds are decreased slightly showing that it is reaching an optimum value which is obtained at (34, 170).

*B*. *Kernel for Dilation and Erosion*

Fixed parameters

a) Threshold values for Canny = 40,197

b) Contour Area: area>avg\_area\*0.2 and area<avg\_area\*4

TABLE I(b)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Kernel for Dilation | Kernel for Erosion | Morphological Image | Final Image | Type of Snake |
| 5,5 | 5,5 |  | Scale Count = 70(overlapping scales) | Rat |
| 7,7 | 7,7 | A black and white image of a black surface  Description automatically generated | Scale Count = 61(extra scales detected) | Rat |
| 9,9 | 9,9 |  | Scale Count = 57 -BEST | Rat |
| 5,5 | 5,5 |  | A close-up of a fish  Description automatically generated  Scale Count = 152 - BEST | Trinket |
| 7,7 | 7,7 | A black and white image of a fish  Description automatically generated | A close-up of a screen  Description automatically generated  Scale Count = 121(Some scales not detected) | Trinket |
| 9,9 | 9,9 |  | Scale Count = 101 (Scales not detected) | Trinket |

From Table I(b), for the processing of rat snake using method 1, we see that the kernel (9,9) performs the best. The rat snake scales are larger and well-defined as compared to those of a trinket snake, therefore a larger kernel provides a higher accuracy. Beyond a (9,9) kernel, the performance starts degrading again. On the other hand, a trinket snake is a snake with much smaller and thinner scales. Therefore, =our algorithm reaches its optimum value for the trinket snake at a (5,5) kernel.

*C. Contour Area*

Fixed parameters

a) Threshold values for Canny = 34,170

b) Kernel for Dilation and Kernel for Erosion = (9,9)

TABLE I(c)

|  |  |  |  |
| --- | --- | --- | --- |
| Lower value | Higher Value | Final Image | Snake Type |
| area>avg\_area\*0.5 | area<avg\_area\*8 | A green outline on a surface  Description automatically generated with medium confidence  Scale Count = 44 | Rat |
| area>avg\_area\*0.3 | area<avg\_area\*2 | A screenshot of a cell phone  Description automatically generated  Scale Count = 48 | Rat |
| area>avg\_area\*0.3 | area<avg\_area\*4 | A screenshot of a computer screen  Description automatically generated  Scale Count = 51 | Rat |
| area>avg\_area\*0.3 | area<avg\_area\*8 | A screenshot of a computer  Description automatically generated  Scale Count = 57 - BEST | Rat |
| area>avg\_area\*0.5 | area<avg\_area\*8 | Scale Count = 104 | Trinket |
| area>avg\_area\*0.2 | area<avg\_area\*2 | Scale Count = 137 | Trinket |
| area>avg\_area\*0.2 | area<avg\_area\*4 | Scale Count = 152 - BEST | Trinket |
| area>avg\_area\*0.1 | area<avg\_area\*9 | Scale Count = 57 | Trinket |

From Table I(c), it is seen that the best performance for trinket snake is given by area > avg\_area\*0.2 and area < avg\_area\*4. When the upper margin for area is too large, for instance with area< avg\_area\*8 and area< avg\_area\*9, parts of the snake that are not contours are being detected as the margin for error is too large. It is unrealistic to accept values 8 and 9 times the average in this case. However when the upper threshold is twice the average and lower threshold is 0.2 times the average it seems to underperform owing to a lack of accurate contours to include. The space for error is practically, given that the images all differ in lighting, brightness, intensity and scale quality. Therefore, a median value was tried which yielded the best results, giving 152 contours which is closest to the actual values. However this is only because the scales on a trinket snake are smaller than those of a snake with larger scales, for instance, the rat snake. In the case of the rat snake the opposite occurs. As the scales are larger and more defined, it is advantageous to keep the higher threshold 8 or 9 times the average as one is more likely to include a scale in them.

TABLE III  
 DIFFERENT STAGES OF METHOD 2 FOR RAT SNAKE: IMAGE SEGMENTATION

|  |  |  |
| --- | --- | --- |
| A close up of a snake skin  Description automatically generatedOriginal Colour | A close-up of a computer mouse  Description automatically generatedGray | A close up of a snake skin  Description automatically generatedInverse Binary Thresholding |
| A close up of a snake skin  Description automatically generated  Segmented Image | A close-up of a snake skin  Description automatically generated  Image Sharpening | A close up of a snake  Description automatically generatedCanny Edge Detection |
| A close up of a black and white image of a cell  Description automatically generatedMorphological Operations | A black and white image of a person standing on a rock  Description automatically generatedBinary Thresholding | A computer mouse in a fish tank  Description automatically generatedOutput Image |

*A. Threshold values for Canny*

Fixed parameters

a) Kernel for Dilation: (9,9)

b) Kernel for Erosion: (9,9)

c) Contour Area: area>avg\_area\*0.09 and area<avg\_area\*1.5

TABLE 2(a)

|  |  |  |  |
| --- | --- | --- | --- |
| Lower Threshold | Upper Threshold | Canny Image | Final Image |
| 20 | 255 | A black and white image of a snake skin  Description automatically generated | A computer mouse on a surface  Description automatically generated  Scale Count = 65(Not scales) |
| 50 | 255 | A black and white image of a snake  Description automatically generated | A computer mouse in a fish tank  Description automatically generated  Scale Count = 48 |
| 80 | 215 | A black and white image of a black and white image of a black and white image of a black and white image of a black and white image of a black and white image of a black and  Description automatically generated | A computer monitor showing a fish tank  Description automatically generated  Scale Count = 10(Scales not detected) |
| 40 | 195 | A close up of a snake  Description automatically generated | A computer mouse in a fish tank  Description automatically generated  Scale Count = 53 - BEST |

*B.* *Kernel for Dilation and Erosion*

Fixed parameters

1. Threshold values for Canny = 45,195
2. Contour Area: area>avg\_area\*0.09 and area<avg\_area\*1.5
3. TABLE 2(c)

|  |  |  |  |
| --- | --- | --- | --- |
| Kernel for Dilation | Kernel for Erosion | Morphological Image | Final mage |
| 5,5 | 5,5 | A close up of a snake skin  Description automatically generated | A computer mouse in a tank  Description automatically generated  Scale Count = 83 |
| 7,7 | 7,7 | A close up of a snake  Description automatically generated | A computer mouse in a fish tank  Description automatically generated  Scale Count = 61 - BEST |
| 9,9 | 9,9 | A close up of a snake  Description automatically generated | A computer mouse in a fish tank  Description automatically generated  Scale Count = 53 |

*C. Contour Area*

Fixed parameters

a) Threshold values for Canny = 45,195

b) Kernel for Dilation and Kernel for Erosion = (9,9)

|  |  |  |
| --- | --- | --- |
| Lower value | Higher Value | Final Image |
| area>avg\_area\*0.4 | area<avg\_area\*6 | A computer mouse in a fish tank  Description automatically generated  Scale Count = 24 |
| area>avg\_area\*0.2 | area<avg\_area\*4 | A computer mouse in a fish tank  Description automatically generated  Scale Count = 42 - BEST |
| area>avg\_area\*0.2 | area<avg\_area\*2 | A computer mouse in a fish tank  Description automatically generated  Scale Count = 25 |
| area>avg\_area\*0.09 | area<avg\_area\*1.5 | A computer mouse in a fish tank  Description automatically generated  Scale Count = 25 |

From Tables 2(a), 2(b) and 2(c) it is evident that the algorithm for Method 2(Image Segmentation) performs very similarly to Method 1. The parameters vary in the same pattern as they do in Method 2, giving similar and quite accurate results.

1. **EXPERIMENTAL SETUP**

Laboratory Setup: There is a glass observation plate over which a camera and light is placed. Snakes of four different species are made to pass over the glass frame. The camera captures the snake's moving body frame by frame and stores the images separately. There is also a camera below the glass plate to capture images of the underside of the snake.

1. **RESULTS AND DISCUSSIONS**

Our final observations and inferences from our prescribed methodology comprise two main components: the scale count and the independent parameters.

1. For a rat snake using methodology 1, we see the result below:

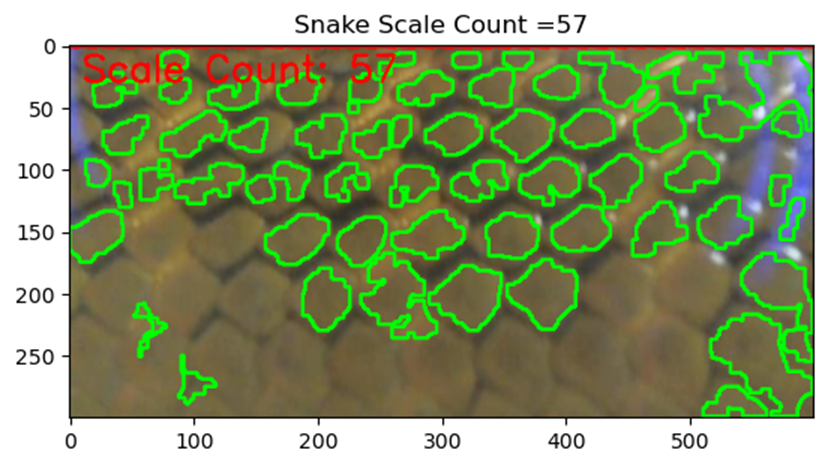


Fig 2. Methodology 1: Actual Count = 62, Predicted Count = 57

The actual number of scales is 62 when counted by hand. Using Method 1 we have received a count of 57 which is well within the 85% accuracy margin. Although the contours do not exactly overlap with all scales, the accuracy is high, showing that necessary features have been included and the parameters have been estimated successfully. There is one region in the final output where there are two contours being detected in one snake scale. This can be fixed by either creating a deep learning model to learn parameters or by implementing a penalty system for misclassification or overclassification.

The parameters of the model are:

Threshold: Lower threshold = 34, Upper Threshold = 170

Kernel Size: 9,9

Area Criteria: area>avg\_area\*0.3 and area<avg\_area\*8

Accuracy: 91%

1. For a rat snake using methodology 2, we see the result below:

A close-up of a scale

Description automatically generated

Fig 3. Methodology 2: Actual Count = 62, Predicted Count = 56

The observations for image segmentation are very similar to those of manual selection edge points.

The actual number of scales is 62 when counted by hand. Using Method 2 we have received a count of 56 which is well within the 85% accuracy margin. Although the contours do not exactly overlap with all scales, the accuracy is quite high, showing that necessary features have been included and the parameters have been estimated successfully. There is one region in the final output where two contours are being detected in one snake scale. This can be fixed by either creating a deep learning model to learn parameters or by implementing a penalty system for misclassification or overclassification. The lower part of the image does not show scales. This is owing to the lighting in the laboratory being directly over this part of the observation glass.

The parameters of the model are:

Threshold: Lower threshold = 40, Upper Threshold = 195

Kernel Size: 7,7

Area Criteria: area>avg\_area\*0.2 and area<avg\_area\*1.5

Accuracy: 90.32%

1. **Conclusions**

The dataset comprises snakes of various species, observed in a laboratory setup. The images all differ in size, lighting and quality. Some snakes have scratches on their scales, making it difficult to accurately count scales. The images that have been used have gone through various steps of preprocessing, such as sharpening, thresholding, segmentation/ cropping and then have been further been passed through a canny edge detector, morphological dilation and erosion, to detect count and draw contours onto the original images with an accuracy of 85%.

1. **FUTUREWORK**

Based on the findings, it is the first step in moving towards the classification of venomous or non-venomous classes of snakes. According to current research, the biological distinction between the two kinds is based on colour pattern and whether it is thinly or thickly spread throughout the body. Another way is to note the shape of the head due to the presence of the venom sack. However, none of these methods are foolproof and require more in-depth research in collaboration with domain experts. We aspire to further this research cause in the due course of time.

Acknowledgement

We would like to take this opportunity to thank our professors – Professor Siddalingaswamy P C, Dr. Muralikrishna SN, and dedicated Ph.D. scholar Mayur Sir for their invaluable guidance and mentorship throughout the project. Their endless support and expertise about the subject helped us in our growth in the subject and the project’s success.

References

[1] I. Kanjanasurat, W. Krungseanmuang, V. Chaowalittawin and B. Purahong, "Egg-Counting System Using Image Processing and a Website for Monitoring," 2021 7th International Conference on Engineering, Applied Sciences and Technology (ICEAST), Pattaya, Thailand, 2021, pp. 101-104, doi: 10.1109/ICEAST52143.2021.9426295.

[2] M. Moorman and A. Dong, "Automated Viral Plaque Counting Using Image Segmentation and Morphological Analysis," 2012 IEEE International Symposium on Multimedia, Irvine, CA, USA, 2012, pp. 157-160, doi: 10.1109/ISM.2012.38.

[3] W. Fankam-ai, P. Rintha, T. Inthajak, S. Kaewsirirung, N. Ketui and K. Homjun, "A Web Platform for Measuring Frog Size and Counting via Contour Detection and Image Segmentation," 2023 20th International Joint Conference on Computer Science and Software Engineering (JCSSE), Phitsanulok, Thailand, 2023, pp. 374-379, doi: 10.1109/JCSSE58229.2023.10202114.

[4] Sacchi, R., Scali, S., Pellitteri-Rosa, D., Pupin, F., Gentilli, A., Tettamanti, S., Cavigioli, L., Racina, L., Maiocchi, V., Galeotti, P., and Fasola, M. (2010). Photographic identification in reptiles: a matter of scales. Amphibia-Reptilia 31, 4, 489-502, Available From: Brill https://doi.org/10.1163/017353710X521546 [Accessed 08 November 2023]

[5] D. Barik and M. Mondal, "Object identification for computer vision using image segmentation," 2010 2nd International Conference on Education Technology and Computer, Shanghai, China, 2010, pp. V2-170-V2-172, doi: 10.1109/ICETC.2010.5529412.

[6] L. Xiaohu and O. Jineng, "Research on steel bar detection and counting method based on contours," 2018 International Conference on Electronics Technology (ICET), Chengdu, China, 2018, pp. 294-297, doi: 10.1109/ELTECH.2018.8401470.

[7] J. Ni, Z. Khan, S. Wang, K. Wang and S. K. Haider, "Automatic detection and counting of circular shaped overlapped objects using circular hough transform and contour detection," 2016 12th World Congress on Intelligent Control and Automation (WCICA), Guilin, China, 2016, pp. 2902-2906, doi: 10.1109/WCICA.2016.7578268.

CONTRIBUTIONS:

|  |  |  |
| --- | --- | --- |
|  | Team Members | Contributions |
| Sreya | Anoushka | Basic Image Processing, Worked on Method 2, Compilation of Report |
|  | Sreya | Basic Image Processing, Worked on Method 1, Tabulation of Observations of varying parameters |
|  | Saksham | Basic Image Processing, Tabulation of Observations, Inputs for Report |
| Anoushka | Anoushka | Basic Image Processing, Worked on Method 2, Analysis of Observations for Report and Compilation of report |
|  | Sreya | Basic Image Processing, Worked on Method 1, Tabulation of Observations of varying parameters, inputs for report |
|  | Saksham | Basic Image Processing, tabulation of observation for analysis, tested different parameters to estimate correct output, Inputs to report |
| Saksham | Anoushka | Basic Image Processing, Worked on Method 2, Analysed Observations for Report |
|  | Sreya | Basic Image Processing, Worked on Method 1, Tabulation of Observations of varying parameters |
|  | Saksham | Basic Image Processing, tabulation of observation, tested different parameters to find correct output, Inputs to report |