Efficient Detection of the Entrance of an Underwater Dock through Object Colour Extraction and Size Estimation

A Report submitted by

Ms. Kulsoom Masood

B.Tech, 2nd Year, Computer Science & Engg.

Aligarh Muslim University

as a

Summer Research Fellowship Program (SRFP)

Indian Academy of Sciences, Bengaluru. SRFP Code: ENGS1891

Supervisor: Dr. Sambhunath Nandy, Chief Scientist

Robotics & Automation, CSIR-CMERI Durgapur.

CSIR-Central Mechanical Engineering Research Institute, Durgapur.
Pin: 713209. West Bengal (State). India.

04th August 2023

Efficient Detection of the Entrance of an Underwater Docking Station through Color Extraction and Size Estimation

A Final Report by

Kulsoom Masood (SRFP Application No. : ENGS1891)

August 4, 2023

1 Introduction

Underwater vision plays an important role in the missions of marine investigations and underwater robotic explorations. Exploring, understanding and investigating underwater activities of images are gaining importance for the last few years. Today, scientists are keen to explore the mysterious underwater world. Considering the evolution of newer avenues and applications, the need for novel image processing techniques for extraction of informative underwater imagery has been on steady rise. Herein, it is worth mentioning that, in several applications, two aspects which received considerable attention in the fields of practice of using underwater vehicles are, *object color detection* and *object size estimation.*

It is known that the optical images captured underwater are usually degraded by absorption and scattering effects that occur during propagation of light.

Images acquired in the marine environment often suffer from haze, low contrast, and color distortion. In particular, the red color gets absorbed first as it has a longer wavelength and lower frequency. Note that, the red color gets absorbed fully at around 5 m depth. This is followed by orange color, which vanishes at a depth of around $7-9$ m. Next, the yellow color gets absorbed around at the depth of $16-18$ m, which is followed by green color at a depth of around $28-30$ m. As a result, underwater images are color cast i.e. they are either dominated by green or blue colors. Figure 1 shows the result of color cast phenomenon in underwater ambience in still waters in a tank during daylight scenario.

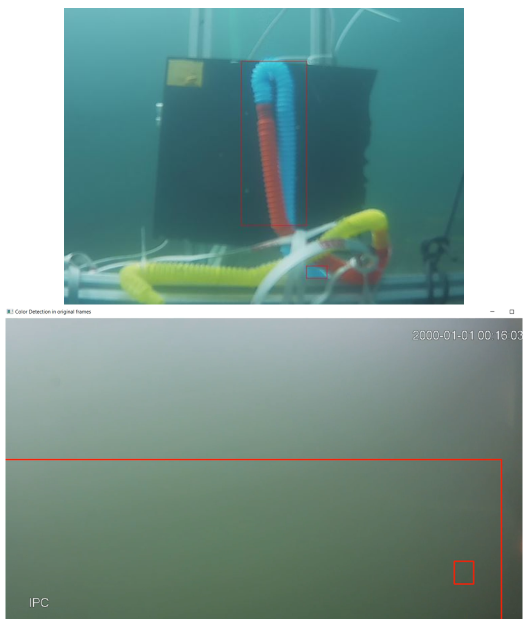

Figure 1: Illustration of color cast in underwater images: Top-Captured in a fresh water tank. Bottom-Captured in turbid water tank.

Since the colors disappear based on the depth, color imbalance is reflected in the images captured in underwater ambience. Needless to mention, the degradation of image quality hinders underwater object color recognition, and in turn the object size estimation, from real field underwater imagery. Consequently, the challenges associated therein are further complicated when the underwater ambience is highly turbid in nature. Thus, enhancing the degraded images is critically important in a variety of vision-based applications in underwater ambience.

In the past, research in image processing was mainly limited to ordinary images with the exception of a few approaches that have been applied to underwater images. For the last few years, a growing interest in marine research has encouraged researchers from different disciplines to explore the mysterious underwater world. A significant amount of literature is available on image processing, 'event detection', 'detection and tracking of objects' and 'feature detection', to mention a few. In particular, power-intensive instrumentation for such high-speed image processing (image quality enhancement, color detection, object recognition, object size estimation) consume large energy of an underwater vehicle and limits their working periods. It follows that for nearly all the missions, the underwater robot, say an autonomous underwater vehicle (AUV), should be attended by a surface vessel or shoreside staff, increasing much cost and inconvenience for missions. In this work, a conceptual application of docking of an AUV is chosen for study the performance image enhancement algorithms in improving the accuracy of color detection and object size estimation in fresh as well as turbid underwater images.

1.1 Motivation for the Present Work

When it comes to deep water applications longer mission times are often limited by the battery endurance. It follows that for extended duration missions, AUV refueling is possible only after retrieving the autonomous vehicle on to research vessel. However, practice of frequent launching and retrieval of autonomous underwater robotic systems is highly risk prone as it could cause severe damage of the system especially while operating in unfavorable sea-states. Besides, on board storage capacity is also associated with limitations of transferring huge data through acoustic channel also pose restrictions on operation time. Above all, with increased operating depths, the time to reach a particular depth and surfacing for charging battery and downloading data also requires a significant amount of time. In the context of underwater docking of an AUV, a common practice is to identify the presence of a docking station in the vicinity of the AUV. This is conceptually realized by arranging light markers in a circular geometry and by extracting the pose of the docking station with respect to AUV from the images captured by the AUV. Figure 2 shows a typial docking system entrance with light markers.

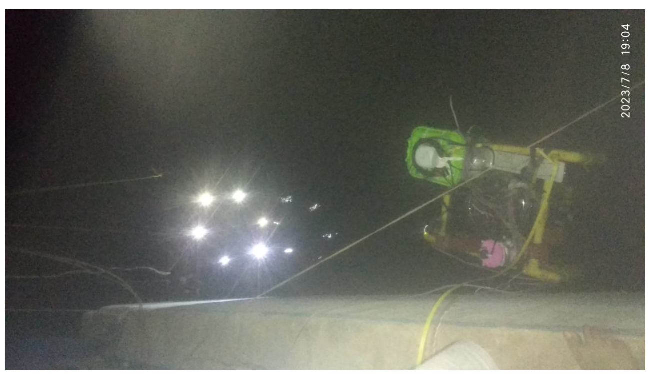

Figure 2: Typical circular docking station with light markers (Captured by a camera attached with an ROV in a large tank at Shallow Basin Facility of CSIR-CMERI Durgapur)

1.1.1 Problems of Interest in the Context of UW Docking

In the context of underwater docking, specifically, following aspects need considerable attention by the design and research engineers:

- **Extraction of Dock's Region of Interest (for Dock Pose Estimation)** To estimate the pose first the light markers need to be detected.

When the ambience in water contains other objects which fall in the field of view of the camera on AUV, the extraction of markers gets difficult Hence it is required that the region of interest of markers is detected. In the present work, an attempt is made to extract the colored markers on a docking station. Herein, the idea is to exploit the presence of the colored markers and their location in the scene to aid navigation algorithm of the AUV to the entrance of the dock. For this, as shown in Figure 3, coordinates of top left and bottom right rectangular boxes are taken and with the help of them, centroid is calculated to find our region of interest. In other word, the problem of detecting colored lights was studied in

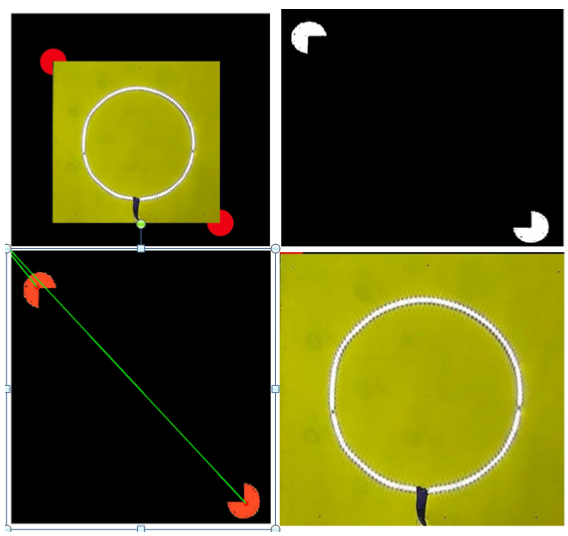

Figure 3: ROI extraction from an image based on color markers. Top Left: Original image of dock with LEDs and colored markers; Top Right: Color detected for ROI extraction; Bottom Left: Centroids of colored markers; Bottom Right: Extracted ROI from image

the context of docking of an AUV. As mentioned earlier, in general, underwater images are color cast. Further, they suffer from various noises, wavelength-dependent absorption etc. As a result, the images acquired from onboard AUV camera are expected to be processed for a desired (color) information extraction.

- **Estimation of Object Size for Extraction of Docks Region of Interest** Size estimation, in the context of computer vision and image processing, refers to the process of determining the physical dimensions or scale of objects in an image or a scene. It involves mapping the size of objects in the image to real-world measurements (e.g., length, width, height) in a specified unit, such as centimeters or meters. This task plays a vital role in numerous applications, including object detection, robotics, autonomous vehicles and industrial automation, to cite a few. Accurate size estimation is essential for making decisions, planning actions, and understanding the spatial relationships between objects in a scene. In the present context, alongside the colored markers, a study was also conducted to compute the sizes of known objects of the dock or in the ambience of the dock. Herein, again the idea is to exploit the knowledge of the known objects in the ambience of the dock, an estimate of which is expected to aid the navigation of the AUV into the dock.

2 A Brief Note on Prominent Image Processing Objectives and their Importance

2.1 Image Quality Assessment

As discussed in Section 1, image enhancement methods are required for underwater images specifically designed to address the challenges posed by the underwater environment, such as color distortion, low visibility, and scattering of light. Following are some commonly used image quality analysis as well as improvement methods for underwater ambience:

- **Color Correction:** Underwater images often suffer from color cast due to the absorption and scattering of light in water. Color correction techniques aim to restore the natural colors by adjusting the color balance and compensating for the dominant color cast.

- **Contrast Enhancement:** Enhancing the contrast of underwater images can improve the visibility of details and structures. Adaptive histogram equalization and local contrast enhancement methods are often employed to enhance the contrast while preserving the local characteristics of the image.

- **Dehazing:** Haze or backscatter caused by suspended particles in the water can significantly degrade the quality of underwater images. Dehazing methods attempt to reduce the effects of haze and improve image clarity by estimating and removing the scattering medium from the image.

- **White Balance Adjustment:** Underwater images often suffer from a loss of color due to the absorption of specific wavelengths of light. Adjusting the white balance can help restore the correct color representation by compensating for the color attenuation at different depths.

- **Image Fusion:** Underwater image fusion techniques combine multiple images of the same scene captured under different lighting conditions or from different viewpoints to create a single enhanced image. Fusion methods can improve the visibility of details and reduce noise and artifacts.

- **Noise Reduction:** Underwater images can exhibit high levels of noise due to low light conditions or limitations of underwater cameras. Noise reduction algorithms, such as adaptive filtering or wavelet denoising, can be applied to reduce the noise while preserving important image details.

- **Light Field Enhancement:** Light field techniques capture both the intensity and direction of light rays in a scene. By utilizing the additional information provided by the light field, it is possible to enhance underwater images, improve depth perception, and reduce the effects of scattering.

- **Image Deblurring:** Underwater images can suffer from motion blur caused by camera movement or water currents. Deblurring methods aim to restore the sharpness and clarity of the image by estimating and compensating for the blur.

2.2 Estimation of the Size of an Object

As dicussed earlier, size estimation, refers to the process of determining the physical dimensions or scale of objects in an image or a scene. It involves mapping the size of objects in the image to real-world measurements (e.g., length, width, height) in a specified unit, such as centimeters or meters. In the present context, estimation of size of objects is expected to improve the reliability in navigation of an AUV towards and into the dock. Typical vision based object size estimation methods are as follows:

- **Stereo Vision and Depth Estimation** One of the simplest methods for size estimation is object detection, where algorithms identify objects of interest and draw bounding boxes around them. The dimensions of the bounding boxes can be directly used to estimate the size of the objects in terms of width and height.

- **Stereo Vision and Depth Estimation:** By using stereo cameras or depth sensors, it's possible to estimate the distance to objects in a scene. Triangulation techniques allow for the estimation of the size of objects in real-world units (e.g., meters or centimeters).

- **Feature-Based Estimation:** Feature detection algorithms like Scale-Invariant Feature Transform (SIFT) or Speeded-Up Robust Features (SURF) can be used to identify key points in an image. By matching these key-points across different images or using reference objects with known dimensions, size estimation can be performed.

- **Reference Object Calibration:** In certain scenarios, a reference object with known dimensions is placed in the scene. By calibrating the camera and measuring the size of the reference object in pixels, it becomes possible to estimate the size of other objects in the scene.

- **Machine Learning Techniques:** Machine learning algorithms, such as regression models or convolutional neural networks (CNNs), can be trained to directly predict the size of objects based on their visual features.

2.3 Edge Detection

Edge detection is used in detecting contours in the image for size estimation. It is a fundamental technique in image processing and computer vision used to identify and highlight the boundaries of objects or regions within an image. The edges represent significant changes in pixel intensity, which often correspond to object boundaries, object contours, or important features in the image. It plays a crucial role in various computer vision tasks, like it helps in identifying objects within an image by highlighting their boundaries or contours, which can then be used for further analysis or recognition. It assists in dividing an image into meaningful regions or objects, which is essential for various computer vision tasks like tracking, object recognition, and image understanding. It provides essential visual features that can be used to represent objects or regions in an image, allowing for extraction of more efficient and meaningful image analysis. It is vital for image compression algorithms that aim to preserve important details while reducing the image size.

- **Gradient-Based Edge Detection:** A sobel "gradient" operator applies convolution masks to compute the image gradient in the horizontal and vertical directions and combines them to obtain the edge strength and direction. A Prewitt operator, is similar to Sobel operator which uses a different set of convolution masks.

- **Laplacian of Gaussian (LoG) Edge Detection:** Applies Gaussian blurring to the image and then uses the Laplacian operator to detect changes in intensity. It enhances edges and suppresses noise.

- **Canny Edge Detection:** A multi-stage algorithm that applies Gaussian blurring, computes gradient magnitude and direction, performs non-maximum suppression, and applies hysteresis thresholding to detect edges. The Canny edge detection algorithm involves several steps: The first step is to apply Gaussian smoothing to the image to reduce noise. This is done by convolving the image with a Gaussian kernel, which helps in removing small variations and unnecessary details. The next step involves calculating the gradient of the smoothed image. The gradient represents the rate of change of intensity in different directions, indicating potential edges. After computing the gradient magnitude and direction, non-maximum suppression is applied. This step helps in thinning the edges by suppressing all the gradient values that are not local maxima along the direction of the gradient. The image is then thresholded with two different threshold values, typically referred to as the high threshold and the low threshold. Pixels with gradient magnitude higher than the high threshold are considered strong edges, pixels with magnitude between the low and high threshold are considered weak edges, and pixels with magnitude below the low threshold are discarded. In this final step, weak edges that are connected to strong edges are retained, while isolated weak edges are suppressed. This is done by tracking along the edges using connectivity information. The output of the Canny edge detection algorithm is a binary image where the detected edges are marked with white pixels and the rest of the image is black. Figure 4 shows the response on a lenna image.

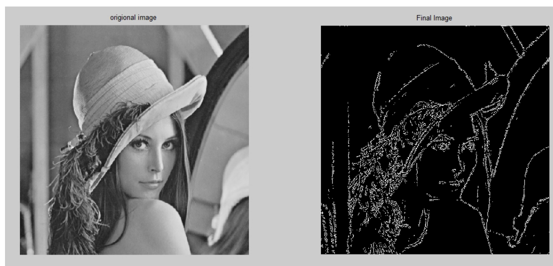

Figure 4: Canny edge detection

3 Experiments, Observations and Inference

In this section, we discuss results obtained from character recognition, color extraction and object size estimation experiments conducted on free space and underwater images. Importantly, we discuss the experimental procedures, observations from experiments. We also discuss potential algorithms for image quality improvement for efficient detection of the region of interest extraction on the underwater dock.

3.1 Optical Character Recognition

Optical Character Recognition (OCR) is a process that converts an image of text into a machine-readable text format. The algorithm uses two necessary libraries, OpenCV and pytesseract libraries to perform text extraction from a webcam feed. It utilizes OpenCV to access the webcam, capture frames, and display the video feed in a window. Note that, it converts the frame from BGR (color) to grayscale because OCR often performs better on grayscale images. Next, the method applies median blurring to reduce noise in the binary image. It then uses pytesseract, a wrapper for the Tesseract OCR engine, to extract text from the frames in real-time. Figure 5 shows the result of OCR from an image captured by a web camera (in free space). As is evident, the most of the text present / detected by the OCR is accurately extracted from the image; however results were not encouraging when tested in water.

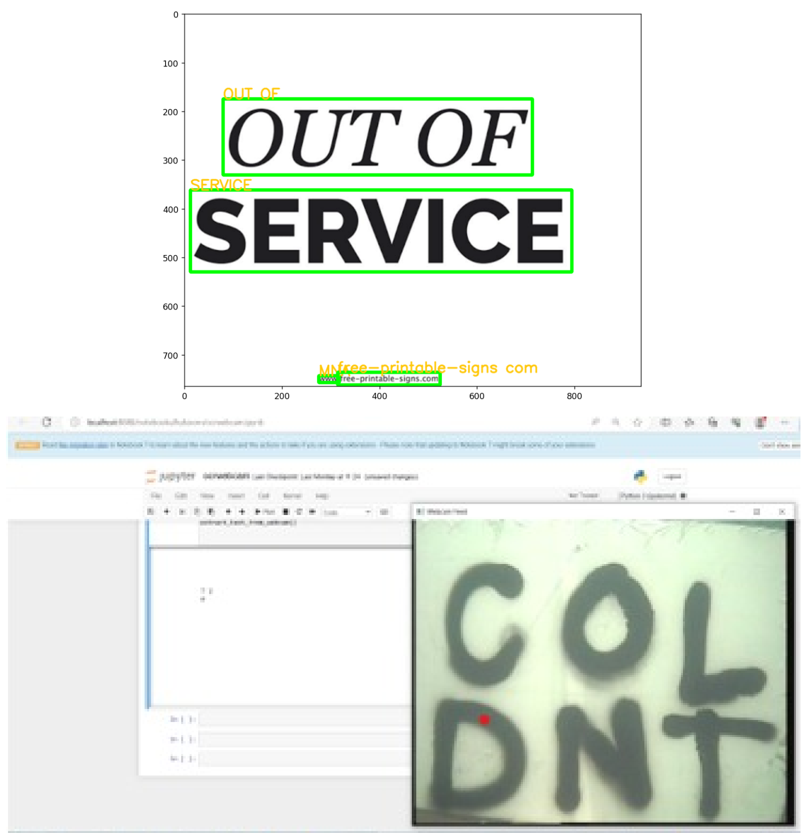

Figure 5: Optical character recognition of text in an images: Top- Freespace; Bottom- Underwater

Difficulties in using OCR underwater OCR for underwater images is a challenging task due to the unique characteristics of the underwater environment. The presence of water introduces various optical distortions, such as scattering, absorption, and color shifts, which can significantly degrade image quality and make text recognition more difficult. Additionally, underwater scenes often contain complex backgrounds, varying lighting conditions, and different types of marine life, which further complicate OCR. Here are some of the key challenges and considerations for OCR in underwater images:

- Image Quality: Underwater images can suffer from low visibility, reduced contrast, and blurring due to light scattering and absorption. These factors can lead to degraded text visibility, making it harder for OCR algorithms to accurately recognize characters.

- Color Variation: Water absorbs and scatters light, causing color distortion in underwater images. This color variation can affect character recognition, as OCR algorithms trained on standard color datasets may not perform well on underwater images.

- Background Noise: Underwater scenes can be cluttered with various objects, corals, and marine life, creating background noise that interferes with text detection and recognition.

- Font and Text Variation: The appearance of text in underwater images may vary significantly in terms of font styles, sizes, and orientations. OCR algorithms need to be robust to handle these variations effectively.

- Limited Training Data: Training OCR models for underwater images can be challenging due to the scarcity of labeled data specific to the underwater domain.

3.2 On Estimation of Size of an Object

In the context of underwater dock ROI extraction, known objects may be placed in the scene. Herein, it is felt that the geometric information of a known object could possibly aid then navigation of the AUV into the dock. Accordingly, existing methods were explored. However in this work, one popular method using a reference object is studied and discussed in detail. In view of its limitations, subsequently, a simple idea is proposed and the performance is evaluated in free space as well as in underwater ambience.

3.2.1 Size estimation with reference object

The experiments for object size estimation were based on computer vision (open cv framework using python). The conventional approach in opencv determine the width and height of objects detected in an image using a reference object whose dimensions are assumed known. The sequence of steps of the method performs are as follows: First the input image is converted to grayscale; next, Gaussian blur and Canny edge detection are performed; subsequently, contours in the edge-detected image are detected. Next, the contours are sorted from left to right, assuming that the leftmost contour is the reference object.

Computation of Size of objects The opencv based approach first removes contours that are not large enough, likely to exclude small noises or artifacts. Next, it calculates the distance of pixels at the two corner points of the reference object. Note, this specifies the dimensions of the reference object in centimeters. It next calculates the ratio of pixels to centimeters ($pixel\_per\_cm$) based on the distance of the reference object. This process iterates through the remaining contours, and for each contour it estimates the width and height of the object in centimeters using the $pixel\_per\_cm$ ratio. Finally, the process draws bounding boxes around the detected objects and displays the image with width and height measurements.

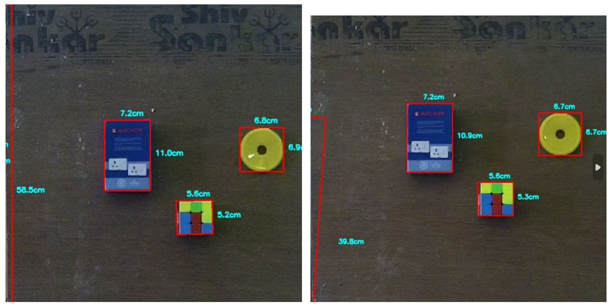

Figure 6: Estimation of size of an object using reference object method at different distances: In free space

Observations from reference object based size estimation The accuracy of size estimation using reference object calibration depends on the precise measurement of the reference object's dimensions. Any errors in the reference object's measurements will propagate to the size estimation of other objects. Additionally, because the algorithm considers the top-left object in the frame to be the reference object, the results will be completely inaccurate if any other object is detected there. Lastly, it is also extremely difficult to ensure the reference object being at the top-left position in the frame, especially for the objects in the vicinity of the reference position whose sizes we need to estimate.

3.2.2 Size estimation without reference object - Proposed Idea

Considering the challenges in reference object based size estimation, next a new idea is proposed which does not require the presence of a reference object. We proposed a methodology for calculating size of objects using the *horizontal* distance between the plane in which the object is lying (or the object) from the camera. The algorithm we used, uses 4 necessary libraries, including scipy.spatial.distance for calculating Euclidean distance, imutils for various image processing functions, numpy for numerical operations, and OpenCV for computer vision tasks.

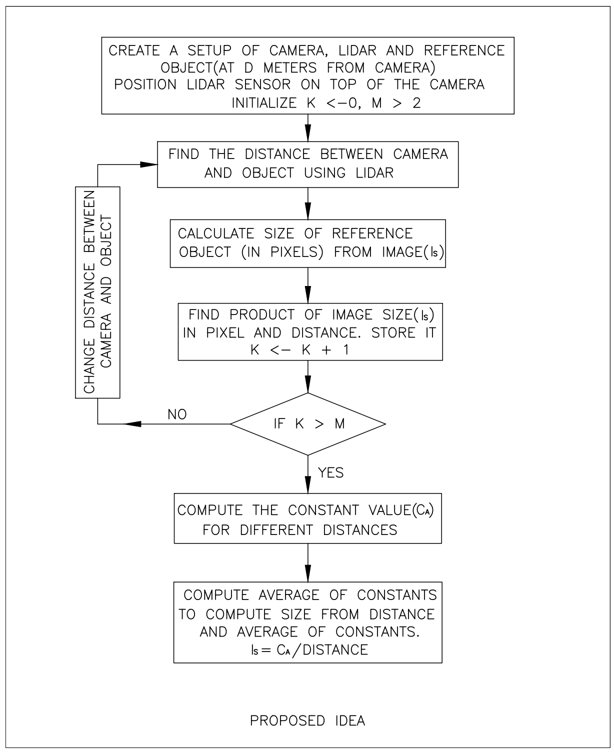

Figure 7: Flow chart of Proposed Idea

The idea is based on the simple fact that the size of the object in the image in terms of pixels is inversely proportional to the distance of the object from the camera. Hence, the multiplication of these two variables is expected to be a constant (or slowly varying about a constant) for any distance and any object for a particular camera. It follows that for any value of distance we can calculate the size of objects in an image in terms of pixels. In the present work, for accurate measurements of distance between object and camera, a LIDAR sensor is positioned above the camera to measure the distance between the object and the camera.[1]

Proposed Methodology: Figure 7 shows a flow chart of the proposed idea. As seen in the flowchart, first we consider an object of known dimension (which is a reference object) and calculate the constant value. Next, it calculates the ratio of image size in pixels to the reference object's size. This ratio allows converting distances measured in pixels to distances in centimeters. The proposed approach next iterates through the remaining significant contours (excluding small ones) and measures the width and height of each object in centimeters based on the previously calculated ratio. It draws bounding boxes around the objects and labels their dimensions on the image. The procedure of experimentation in a real scenario is provided in Figure 8.

Challenges and Considerations in Size Estimation Problem:

- Perspective Distortion: Size estimation can be challenging when dealing with objects at varying distances from the camera, as perspective distortion affects their apparent sizes.
- Scale Variability: Objects may appear at different scales in the same scene, requiring robust algorithms to handle scale variations.
- Accuracy and Precision: The accuracy of size estimation depends on factors like image resolution, camera calibration, and the quality of the detection or feature extraction algorithms.

[1] A LiDAR (Light Detection and Ranging) sensor is an active remote sensing technology that uses laser light to measure distances and create 3D maps of its surroundings. These sensors emit rapid pulses of laser light (typically in the near-infrared or visible spectrum) towards the surroundings. These laser pulses are emitted in all directions, and the sensor measures the time taken for the laser light to travel from the sensor to the target and back. When the laser light encounters an object or a surface, a portion of it gets reflected back towards the LiDAR sensor. The sensor's receiver detects the returning laser pulses, which are known as "echoes." By measuring the time elapsed between the emission of the laser pulse and the reception of its echo, this sensor can calculate the distance between itself and the object or surface.

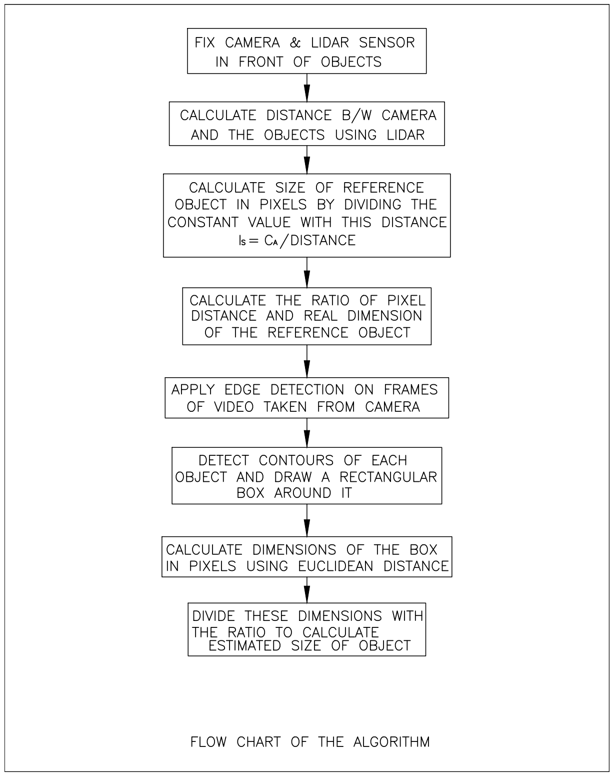

Figure 8: Flow chart of Experimental Procedure

- Real-World Conversion: Converting estimated sizes from pixels to real-world units requires knowing the physical dimensions of reference objects or using additional depth information.

Results from Size Estimation Experiments Using Proposed Idea For experiments with size estimation, a plywood based setup was made with an underwater camera and an object Rubiks cube both at a horizontal distance of 0.85 meters. Figure 9 shows the setup made for experimentation in free space as well as fresh and turbid waters. Note that, for underwater case, since the it was not possible to submerge a normal laser, the actual distance between camera and the object / plane of the object was hard-coded in the program.[2] However, objects sizes and the distances were varied during experiments.

Figure 9: Experimental setup using underwater camera and an object (Rubiks Cube) of known dimensions

Figure 10 shows the size estimation results and the contours of the objects detected (on the right) for free space as well as in underwater ambience. As seen in the free space case, several objects were seen in the scene and sizes of all the objects were estimated and shown in the figure. On the other hand, in the underwater case, as expected, not many external objects other than the object are seen in its field of view by the camera. As a result, the detection of the object as well as its size estimation were fairly accurate.

Important Observation One important observation from size estimation experiments was that the size of the object is smaller to that of object in free space. A careful observation of the sizes of all the known objects in free space and

[2] In a real life scenario, the distance would be obtained by underwater robot using an underwater optical laser.

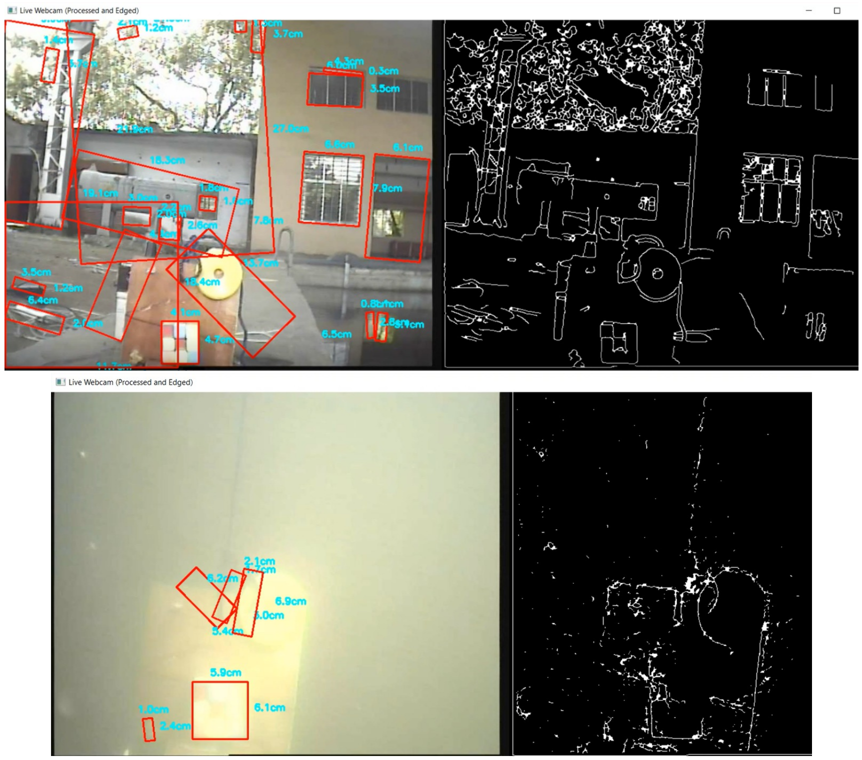

Figure 10: Contour detection and Size estimation of object (Rubiks Cube) of known dimensions: Top: Free space. Bottom: Underwater ambience

the underwater images revealed that the change in refractive index from a glass medium to that of the water apparently reduces the sizes. Accordingly, when a factor of refractive index of water (1.33) was multiplied with the estimated sizes, the dimensions of the objects were close to those of the objects in free space. It follows that size estimation based ROI extraction is indeed a viable alternative for detection of the entrace of an underwater docking station.

3.3 On Image Quality Improvement for Dock's Region of Interest (Entrance) Extraction

For image quality assessment, in the present work we consider target object color detection accuracy. In other word, the objective is to accomplish target object color detection with the usage of image quality improvement algorithm.

3.3.1 RGB to HSV Color Space Transformation for Color Extraction

In the present work, we use the popular color coordinate transformation methodology for color extraction, RGB-to-HSV conversion. Details and the mathematical expressions of the color coordinate transformation procedure are provided in the following discussion. First define,

$$M = \max\{R, G, B\} \text{ and } m = \min\{R, G, B\}.$$

The RGB to HSV transformation expressions are given as:

$$\begin{aligned} &V = M/255 \\ &S = 1 - m/M \text{ if } M > 0 \\ &S = 0 \text{ if } M = 0. \end{aligned} \tag{1}$$

Also, the hue component (H) is defined by the equations

$$\begin{aligned} cos(H) &= (R - 0.5G - 0.5B)/K, \text{ if } G \geq B \\ &= 360 - (R - 0.5G - 0.5B)/K, \text{ if } G < B \end{aligned} \tag{2}$$

where $K = \sqrt{R^2 + B^2 + G^2 - RG - RB - GB}$ and H is computed in degrees using inverse cosine of the expression evaluated on the right hand side. The HSI color model represents every color with three components: hue (H), saturation

(S), and intensity (I). The Hue component describes the color in the form of an angle between [0,360] degrees. The Saturation component describes how much the color is diluted with white light. The range of the S varies between [0,1]. The Intensity range is between [0,1] and 0 means black, and 1 means white. The RGB color model is the most common color model used in Digital image processing and openCV. The color image consists of 3 channels, one channel each for one color. Red, Green, and Blue are the main color components of this model. All other colors are produced by the proportional ratio of these three colors only. 0 represents black and as the value increases the color intensity increases.

Using the said RGB to HSV conversion methodology, experiments were conducted on synthetic images using OpenCV library based color detection algorithm was performed over a live video stream acquired from webcam. The program was targeted to detect objects in the specified color range and to draw bounding rectangles around them. The program begins with importing the necessary libraries, including OpenCV and NumPy. It then defines the color range for detection using the lower and upper HSV (Hue, Saturation, Value) values.

3.3.2 Results of Color Extraction in Free Space

As a first step, color detection is performed on images read using OpenCV. Subsequently, frames from video acquired from the laptops webcam are extracted and the color detection algorithm is applied on each and every frame. From Figures 11 and 12 it is observed that detection of individual colors as well as multiple colors is fairly accurate when it is performed in free space.

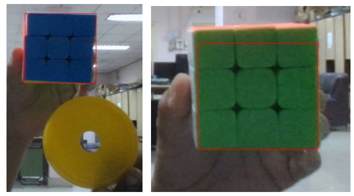

Figure 11: Detection of blue (left) and green (right) color in free space

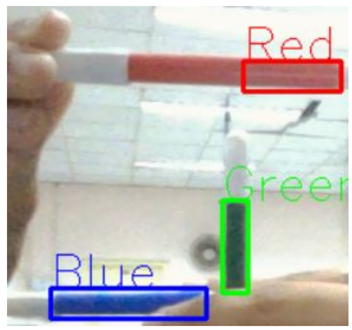

Figure 12: Detection of multiple colors at a time in free space

3.3.3 Results of Color Extraction in Underwater Ambience

Considering the success of free space experiments, next, a setup is made with an inhouse made underwater IP camera as well as commercial grade off the shelf available analog underwater camera. Note that, for underwater camera, an analog to ethernet converter (Make: Moxa, Model: VPort-461) was used to acquire underwater images. Figure 13 depicts the setup made for experimentation. Experiments were conducted in water in a confined space (small tank) located near Robotics & Automation Division, CSIR-CMERI Durgapur.

From Figures 14-16, it is observed due to color cast, most of the images visually seem to be bluish green in color. It is also observed that yellow, blue and green colors are detected in underwater ambience in those figures, respectively.

Figure 13: Setup made for underwater color detection using an underwater IP camera

Figure 14: Detection of Yellow Color in Underwater Ambience

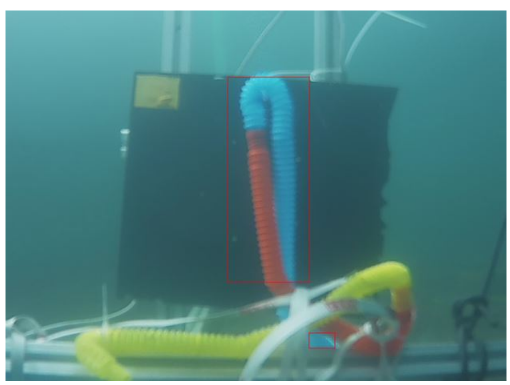

Figure 15: Detection of Blue Color in Underwater Ambience

3.4 Application of Image Enhancement Techniques for Improved Color Detection in Underwater Ambience

It may observed that since the experiments in Figure 14-16 were conducted during dominant daylight conditions, all the objects with same / similar color

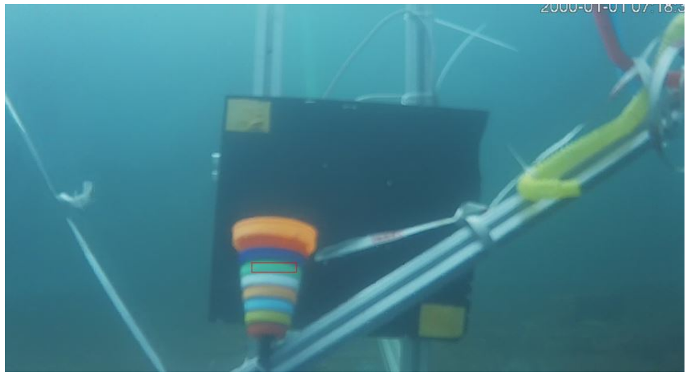

Figure 16: Detection of Green Color in Underwater Ambience

are not detected efficiently. It is felt that image enhancement techniques would improve quality of images and hence improves the accuracy of color detection. In the following, we first provide a brief note on popular image enhancement algorithms. Subqsequently, we provide experimental results.

3.4.1 White Balance Transform

White balance adjustment is a process used in photography and digital imaging to correct the color temperature of an image and ensure that white objects appear truly white, regardless of the color of the light source used during image capture. The purpose of this is to correct the color balance of an image to make it appear more natural and visually appealing. It adjusts the intensities of the red, green, and blue channels in an image to make sure that white objects appear truly white. The method used here is the region of interest (ROI) method. It calculates the sums of the red, green, and blue color channels in a small region of interest (ROI) from the input image. Based on the sums, it computes the color balance ratios and applies them to the entire image to correct the white balance.

3.4.2 Gamma Correction

Gamma correction is a technique used to adjust the brightness and contrast of images by modifying the intensity values of the pixels. To overcome the difficulties with underwater imaging, gamma correction can be used to change the brightness and contrast of the images. These images typically require specific gamma values depending on the water conditions and the imaging equipment used. Since the light absorption and scattering properties can vary in different underwater environments, it's essential to experiment with different gamma values to find the one that provides the best enhancement for a particular image or dataset. Gamma correction is a nonlinear operation that adjusts the brightness levels of an image . In the present work, we performed gamma correction on a grayscale image. In this implementation, gamma value of 0.7 is used to slightly darken the image.

3.4.3 Histogram Equalization

Histogram equalization is a technique used in image processing to enhance the contrast of an image by redistributing the intensity levels. The goal is to spread out the intensity values across the entire range, making the darker areas lighter and the brighter areas darker. It can be applied to underwater images to improve their visibility and enhance details in low-contrast or hazy conditions. Underwater images often suffer from poor visibility due to factors such as light scattering and absorption, which can lead to reduced contrast and color distortion. The formula for equalization is: New intensity = round((L - 1) * CDF(original intensity)) where L is the number of intensity levels (usually 256 for an 8-bit grayscale image), and CDF(original intensity) is the normalized CDF value corresponding to the original intensity.

Results from Experiments in fresh and turbid waters In this work, we considered white balance enhancement approach for improving the quality of the image and in turn the color detection accuracy. Experiments were conducted in fresh as well as turbid waters. Figures 17-19 provide a comparison of enhancement of color images in fresh water conditions. As seen in the figures, red and yellow colors are successfully detected in turbid water conditions even in daylight conditions.

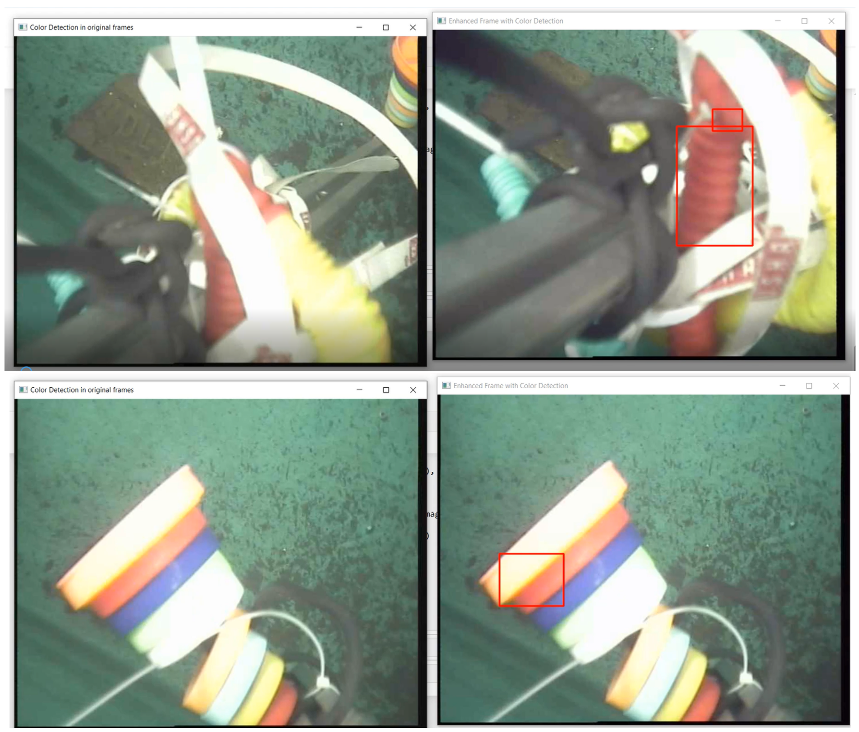

Figure 17: Image enhancement and detecting red color

Figures 21-20 provide a comparison of enhancement of color images in turbid water conditions. Once again, it is evident that while color detection was unsuccessful in unenhanced images, both red and blue colors are successfully detected in turbid water conditions.

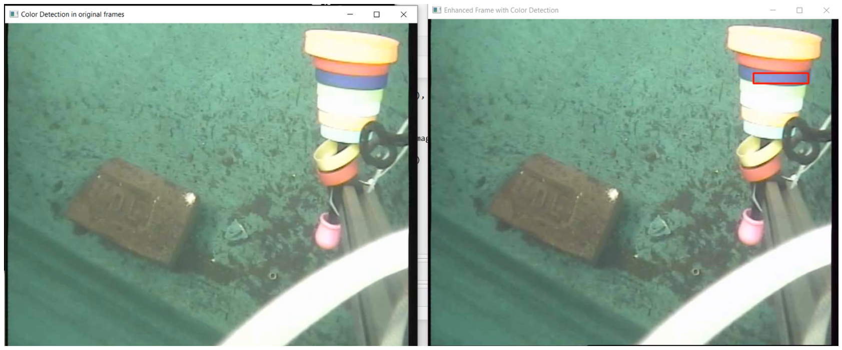

Figure 18: Image enhancement and detecting blue color

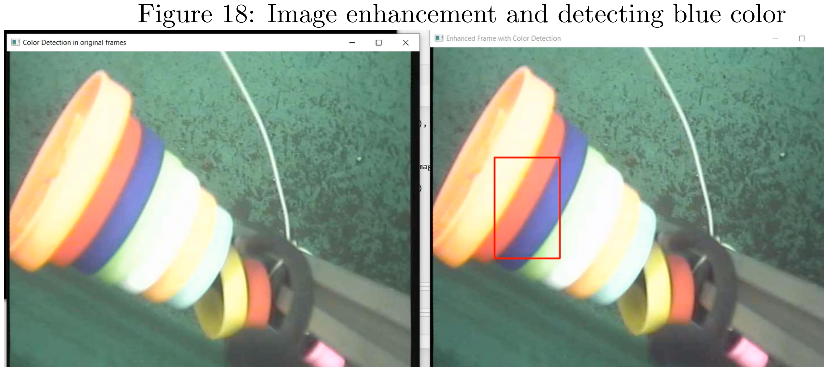

Figure 19: Image enhancement and detecting yellow color

4 Conclusion on Work Done and Further Work

Autonomous docking is an extremely challenging problem for an underwater vehicle to accomplish. The primary objective of an AUV targeting to dock is capable of detecting the dock and subsequently enter the dock. In this work, first the performance of popular color detection scheme (RGB to HSV transformation) is studied over images acquired from different camera in different lighting conditions. It is observed the color detection algorithm works fairly accurate in free space. Specifically, it is noted that images as well as running videos with dynamic color changes, objects of different colors have been successfully identified, even with multiple colors in the image. On the other hand, when experiments were conducted in underwater ambience with a setup made

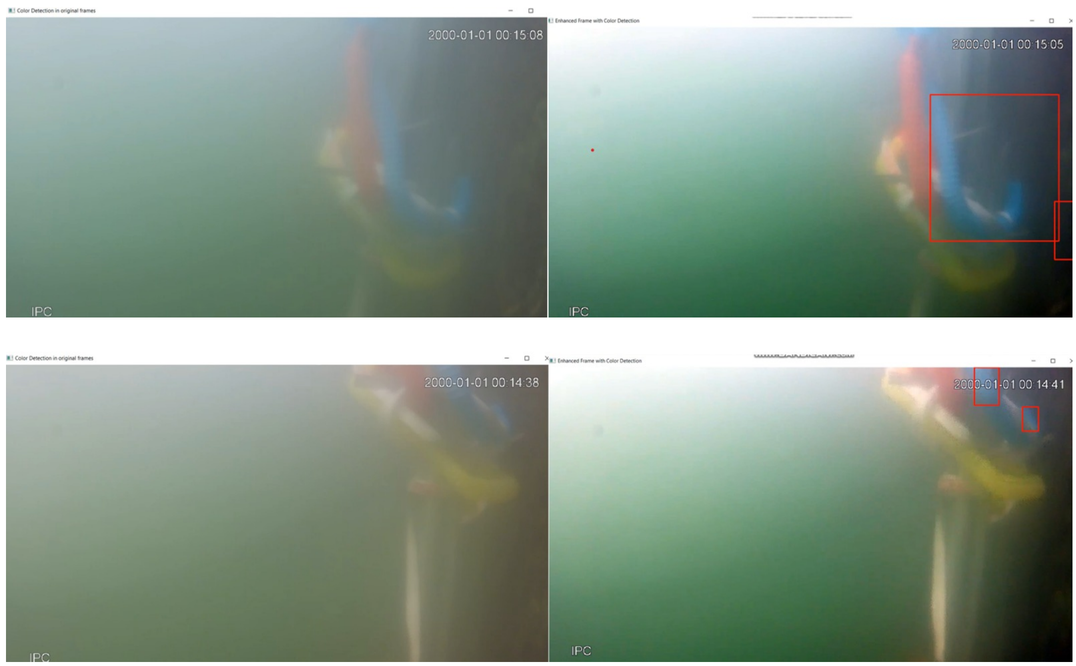

Figure 20: Blue color enhancement of underwater images in turbid conditions

Figure 21: Red color enhancement of underwater images in turbid conditions

for the purpose, it is observed that due to absorption and scattering effects that occur during light propagation. As a result, the color coordinate transformation based object color detection was capable of underwater for blue and green colors. An important observation from the experiments was that, enhancement of images greatly improves the accuracy of color detection. In view of the obtained results, we opine that vision based techniques using neural network models may be used for improving the color detection and in turn efficient dock entrance detection.

Another aspect which was studied is possibility of object size estimation in underwater ambience. In case of object size estimation, it was observed that the conventional approach requires placing a known object in top left side of the scene (or the captured image). Since such an exercise is difficult practice, in this work a simple idea is proposed which does not require any reference object in the scene. Experiments with the proposed idea in free space as well as underwater ambience indicated that the estimates of the sizes / dimensions of the objects are fairly accurate.

Lastly, in view of the success of experimental results and the observations therein, there is a strong motivation to continue further work in this area, especially, optical character recognition, color detection and size estimation for underwater dock entrance detection application under consideration.

5 Acknowledgments

I wish to place on record my sincerest good wishes to all those who have helped me in my pursuit of sincere efforts for Summer Research Fellowship Program as a part of Indian Academy of Sciences (IAS), Bengaluru. First and foremost, I wholehearted thank Indian Academy of Sciences for offering me this opportunity. I thank Dr. Biplab Choudary, Principal Scientist for coordinating and supporting me all through the program. In that order I thank Dr. Sambhunath Nandy, Chief Scientist and my supervisor for his invaluable support throughout my internship. I also thank Dr. Siva Ram Krishna Vadali, Senior Principal Scientist, CSIR-CMERI Durgapur, Dr. Srinivasan Aruchamy, Principal Scientist, CSIR-CMERI Durgapur for their techno scientific contributions towards understanding the problem and solving. I also thank Jyotiroy Karmakar Ji, Chandan Ji, Pintu da, Ms. Sayanica Di, Adnan, Suraj. Indeed it was a great opportunity for me to spend at CSIR-CMERI Durgapur, as a part of my academic schedule. I will to cherish my memories at Robotics and Automation Division, CSIR-CMERI Durgapur in times to come.