*International Journal of Marine Engineering Innovation and Research, Vol. 7(1), Mar. 2022. 18-25*
*(pISSN: 2541-5972, eISSN: 2548-1479)* 

# Prototype of Bridge Navigational Watch Alarm System Equipped Obstacle Warning System Based on Image Processing and Real-Time Tracking

Iskandar[1], Dewa Pamungkas[2], Imam Sutrisno[3], Afif Zuhri Arfianto[4], Ari Wibawa Budi Santosa[5], Iie Suwondo[6]

(Received: 13 February 2022 / Revised: 10 March 2022 / Accepted: 24 March 2022)

***Abstract*— Ships are sea transportation that is often used in Indonesia. However, who would have thought that the transportation would become a case because of the frequent occurrence of accidents. This has been proven from data from the KNKT (National Transportation Safety Committee) which noted that from 2012 to 2017 there had been an increase in accidents in the waters. In fact, according to the National Search and Rescue Agency (Basarnas) in 2020 there have been 878 incidents with victims reaching 4658 people. In this final project, the author makes a prototype of BNWAS (Bridge Navigational Watch Alarm System) equipped with obstacle warning using image processing with the otsu method strengthened by thresholding-based segmentation with inverse technique (TsTN), distance detection using the triangle similarity method, real-time tracking with GPS (Global Positioning System) and the entire system can be observed on the Android application. The Final Project performs several analyzes including performance analysis by calculating the accuracy of the BNWAS alarm system, image detection accuracy, distance detection accuracy, GPS accuracy, overall system testing accuracy and packet loss. The accuracy of each system is very good because the error is below 2%, while for overall system testing has a very good performance with a delay of 179.8 ms and 0% packet loss.**

*Keywords*— BNWAS, collision, GPS, Otsu method, triangle similarity.

## I. Introduction

Ships are sea transportation that is often used in Indonesia. However, who would have thought that the transportation would become a case because of the frequent occurrence of accidents. This has been proven from data from the KNKT (National Transportation Safety Committee) which noted that from 2012 to 2017 there had been an increase in accidents in the waters. In fact, according to the National Search and Rescue Agency (Basarnas) in 2020 there have been 878 incidents with victims reaching 4658 people [1].

According to data from the National Transportation Safety Commission, the majority of these accidents were caused by explosions or fires originating from damage to the ship's system, in addition to other worrying things, accidents occurred due to collisions which were the third largest cause of ship accidents. This is certainly very worrying considering that Indonesia is a maritime country. What is more worrying is happening to our fishermen. The position of accident risk is in an unacceptable risk position, meaning that efforts to reduce the risk of accidents are needed within one year. This is because the data shows that 115 people die per 100,000 crew members per year and is still higher than the world-class fatal accident rate on fishing vessels, which is 80 people per 100,000 crew members. This is because the fishing boat security system is very lacking, it can even be said that 70.31% is not feasible [2].

Therefore, in this paper, the author makes a system concept to reduce the number of ship accidents due to collisions, namely by complementing the BNWAS system, which is coupled with an Image Processing-Based Obstacle Warning system with the Otsu method and real time tracking on ships. It is hoped that this BNWAS prototype will also assist fishermen in implementing safety technology on ships, especially for 150 Gross tonnage (GT) fishing vessels in accordance with standardization.

The system in this paper will implement the otsu method with the addition of TsTN to detect the presence of obstructions in front of the ship, as well as GPS-based tracking with the IoT System, where when detected and there are obstacles on the ship can be control on Android smartphones based on cloud firestore. The system will also be equipped with a tracking path through the google maps API that can be accessed on Android smartphones and information on the distance of obstacles.

BNWAS was introduced through amendments to Safety Of Life At Sea (SOLAS) 1974 CHAPTER V Rule 19, namely with the approval of members of the International Maritime Organization (IMO) at the 86th Maritime Safety Committee (MSC 86) session as outlined in MSC Resolution No. 282 (86) on dated 5 June 2009. SOLAS Chapter V of Regulation 19 states

Iskandar is with Studi Nautika, Politeknik Pelayaran Sorong, Sorong, Indonesia. E-mail: iskadarbp2ipsby@gmail.com

Dewa Pamungkas is with Politeknik Perkapalan Negeri Surabaya, Surabaya, Indonesia. E-mail: dewapamungkas@student.ppns.ac.id

Imam Sutrisno is with Politeknik Perkapalan Negeri Surabaya,Surabaya, Indonesia. E-mail: imams3jpg@yahoo.com

Afif Zuhri Arfianto is with Politeknik Perkapalan Negeri Surabaya, Surabaya, Indonesia. E-mail: afif@ppns.ac.id

Ari Wibawa Budi Santosa is with Universitas Diponegoro, Semarang, Indonesia. E-mail: arikapal75@gmail.com

Iie Suwondo is with Politeknik Pelayaran Surabaya, Surabaya, Indonesia. E-mail: iiesuwondo@poltekpel-sby.ac.id

that all vessels 150 GT and above must install BNWAS. BNWAS is equipment installed on the bridge and is an alarm system for the navigational watch service on the bridge to monitor bridge activity and detect operator errors that can cause accidents at sea. The purpose of the BNWAS is to monitor activities on the bridge and detect operator errors [3]. BNWAS will automatically activate if the ship's rudder is placed in the "autopilot" position (automatic steering function). The minimum requirement for BNWAS in accordance with the provisions of the IMO is to have a dormant stage and three alarm stages, except on passenger ships and fishing vessels, the 2nd stage alarm may be removed.

Although basically BNWAS is a tool that is used to facilitate the navigation system on ships and can prevent the negligence of the duty officer. But it is often neglected and not used because of its high cost or lack of awareness. Especially for fishermen who often ignore this minimum safety equipment where those who have met the minimum safety equipment at the research location have only reached 29.69%, as many as 70.31% have not been equipped with safety equipment according to the minimum requirements [2]. Economics is the main reason for this because navigational security tools are often expensive. Therefore, the BNWAS system used in this final project will use 2 alarm systems in accordance with regulations and complete features so that fishermen only need to buy one tool and can fulfill all their needs. Thresholding is an important technique in image segmentation applications. The basic idea of thresholding is to choose the optimal gray-level threshold value to separate the object of interest in the image (object of interest) from its background (background), based on its gray-level distribution. Since humans with the naked eye can easily be differentiated by complex objects and backgrounds, image thresholding is a difficult task to do, with the aim of separating the two [4].

The Otsu method is usually used to calculate the threshold value automatically, with the aim of separating the object in question (object of interest) from its background. This histogram-based method divides the gray level of an image into two classes. The gray level consists of various levels of gray that vary from black at the weakest intensity, to white at the strongest intensity, which ranges from 0 to 255.

The threshold value is then used to isolate the area of interest (object area) from the background by converting the grayscale image (image) into a binary image. A binary image consists of black and white pixels, where pixels with a gray level greater than the threshold value are considered white and all other pixels are considered black. The resulting binary image is a segmented image, where "value 1" (white) represents the object of interest and "value 0" (black) represents the background. However, in some cases it may be found that the image does not have the right shape to represent the area in question, this is due to a mismatch of the threshold value which is calculated automatically. This situation shows that the existing Otsu method, unable to segment the natural image correctly. Therefore, modification of the threshold values is needed to extract only certain areas (area of interest) [5].

## II. METHOD

This section will describe the TsTN method, which is an improved segmentation technique for natural images. Improvements to the segmentation technique are carried out by modifying the threshold-based segmentation algorithm combined with the inverse technique. This method is referred to as the TsTN method. The method of the improved thresholding-based segmentation technique is divided into three main steps, namely initializing the threshold value, converting the grayscale image into a binary image, and modifying the threshold value which is then used in the image segmentation process [5]. In Figure 1 describes the workflow of the system to be created. The system will be started by pressing or turning on the device, then starting to work where there are three blocks, namely the GPS block, the alarm system block and the camera block. In the GPS block the system will take the coordinates and display them in real-time on the application, while in the alarm system block which will start with setting the time, if the time setting is met, the alarm will sound, as well as on the camera if it detects an obstacle it will turn on a warning alarm and the alert will be silent when the silent button is pressed. And when the camera detects an obstacle, the distance of the obstacle will be raised as well.

### *A. Triangle Similarity for Distance Calculation*

To determine the distance from the author's camera to a known object or marker, the author will use a similarity triangle. The similarity triangle goes like this: A marker or object of known width W. Then place this marker some distance D from the camera. Image of the object using the camera and then measuring the apparent width in pixels P . This allows us to obtain the perceived focal length F from the camera and its equation [6]

.

$$F = (P \times D)/W \quad (1)$$

Then to calculate the distance between the camera and the object can be calculated using the following equation: [6].

$$D' = (W \times F)/P \quad (2)$$

### *B. System Work*

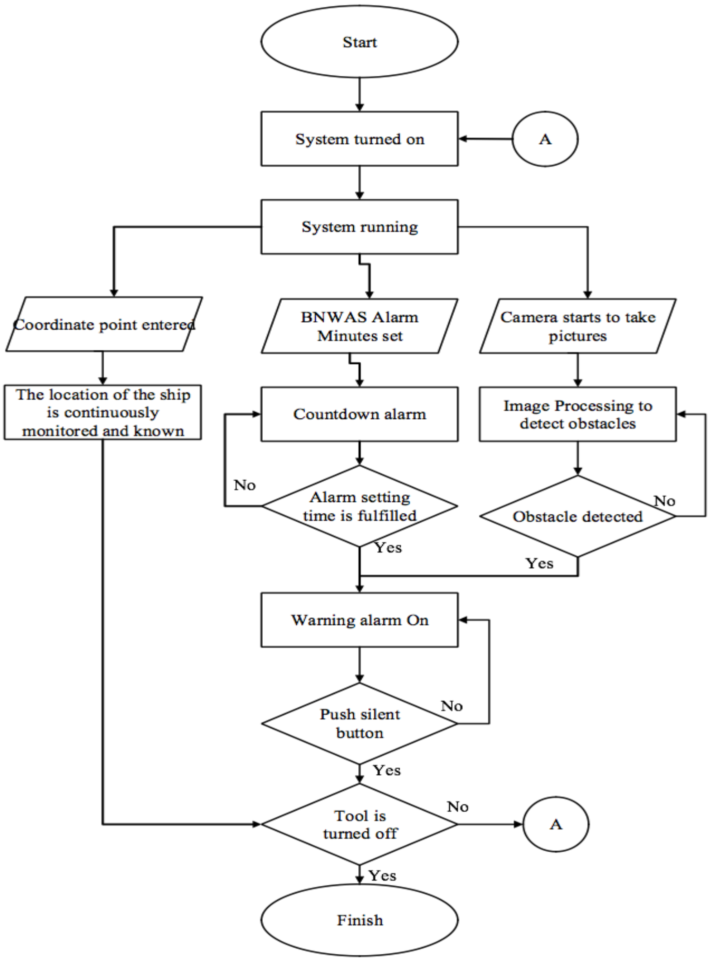


**Figure. 1.** System workflow diagram

*C. Tool Design*

Figure 2 is a tool design that will be made by the author where there are several main parts including:

1. Camera connected by USB.
2. Controller box containing raspberry pi 4, ESP32, GPS, OLED LCD, buzzer, keypad, and buttons.
3. Android Smartphones.

The connection between the hardware can be seen in Figure 2.

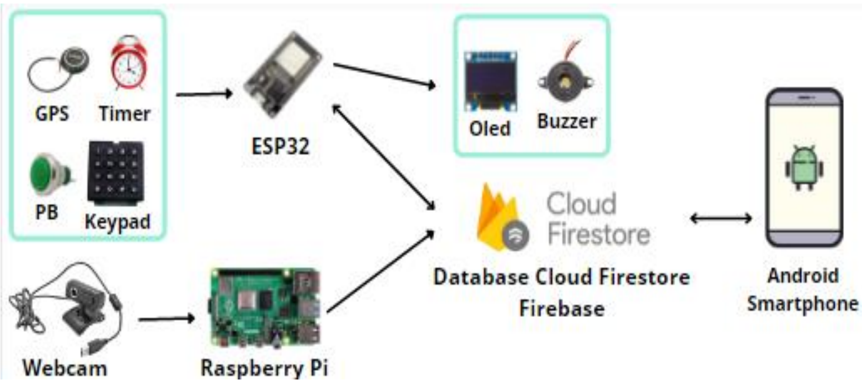


**Figure. 2.** Hardware Connection

## III. Results and Discussion

*A. Alarm System Test*

In this first test, the time delay system programmed on the ESP32 will be tested, then compared with the stopwatch on the Android smartphone. This test will look for absolute error and relative error to determine the accuracy of the programming system and reading system of the microcontroller on the ESP32. [7] The test will use a multiple of 2 minutes, which is often used in BNWAS. Table 1 shows the results of the test.

TABLE 1.
ALARM SYSTEM TEST RESULTS

| Program Time (seconds) | Stopwatch time (seconds) | Absolute error (seconds) | Relative Error (%) |
|---|---|---|---|
| 120 | 121 | 1 | 0.82 |
| | 121 | 1 | 0.82 |
| | 122 | 2 | 1.65 |
| 240 | 244 | 4 | 1.63 |
| | 245 | 5 | 2.04 |
| | 244 | 4 | 1.63 |
| 360 | 366 | 6 | 1.63 |
| | 365 | 5 | 1.36 |
| | 367 | 7 | 1.90 |
| 480 | 487 | 7 | 1.43 |
| | 489 | 9 | 1.84 |
| | 487 | 7 | 1.43 |
| 600 | 610 | 10 | 1.63 |
| | 610 | 10 | 1.63 |
| | 610 | 10 | 1.63 |
| | 610 | 10 | 1.63 |
| 720 | 732 | 12 | 1.63 |
| | 732 | 12 | 1.63 |
| | 731 | 12 | 1.50 |
| | Total Errors | | 1.54 |

*B. GPS System*

This second test will test the accuracy of the GPS programming system on the ESP32 using the Ublox Neo M8N GPS hardware. Table II is a series of tests used for testing. The test will take latitude and longitude data and then compare it with that on google maps on an android smartphone. Tests will be carried out in 10 different places, later will be compared, and produce differences. Here are the results of the test.

TABLE 2.
GPS SYSTEM TEST RESULTS

| On Google Maps | | On Prototype | | Dist. (m) |
|---|---|---|---|---|
| Latitude | Longitude | Latitude | Longitude | |
| -6.823490 | 107.639505 | -6.823481 | 107.639510 | 1,14 |
| -6.846577 | 107.598730 | -6.846595 | 107.598709 | 3,06 |
| -6.818412 | 107.624260 | -6.818392 | 107.624285 | 3,54 |
| -6.817637 | 107.622842 | -6.817661 | 107.622862 | 3,46 |
| -6.863967 | 107.594595 | -6.863987 | 107.594582 | 2,64 |
| -6.838504 | 107.598281 | -6.838519 | 107.598299 | 2,59 |
| -6.843537 | 107.598583 | -6.843549 | 107.598597 | 2,04 |

| | | | | |
|---|---|---|---|---|
| -6.863594 | 107.594413 | -6.863574 | 107.594426 | 2,64 |
| -6.858265 | 107.595176 | -6.858288 | 107.595194 | 3,23 |
| -6.868626 | 107.593605 | -6.868601 | 107.593627 | 3,69 |
| | | | **Average** | **2,8** |

### C. *Image Processing Test*

Based on a visual comparison of images, the output image using the otsu and TsTN methods as shown in the image does not show a significant difference except for the yellow image as shown in Figure 3. In Figure 3 the Otsu method the background looks random and untidy even though the object is fixed. The object still visible, but if left unchecked it will be problematic or cause errors when used for object recognition later.

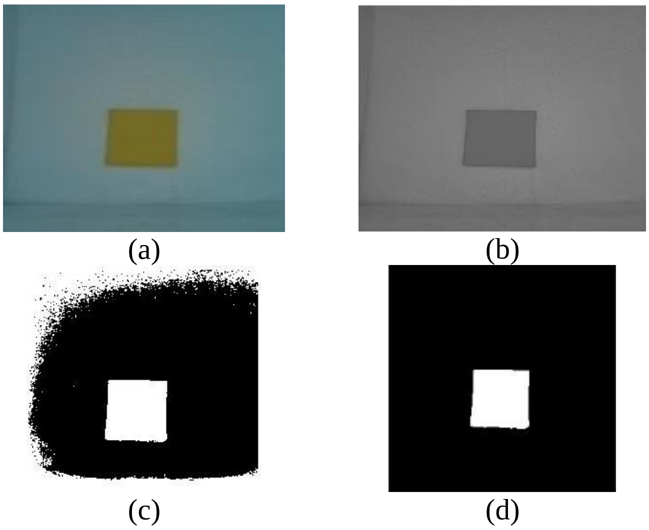


**Figure. 3.** Changes in the Image of Yellow Color Objects; (a) Original image (b) Greyscale image (c) Otsu image (d) TsTN image

TABLE 3.
MSE AND PSNR TEST RESULTS

| Object Name | Color | Threshold (T) | Otsu | | TsTN | |
|---|---|---|---|---|---|---|
| | | | MSE | PSNR | MSE | PSNR |
| Object 1 | Blue | 0.4569 | 0.000576 | 80.5606 | 0.000512 | 81.0721 |
| | Green | 0.4392 | 0.1057 | 57.9242 | 0.1059 | 57,915 |
| | Yellow | 0.4667 | 0.2673 | 53.8951 | 0.2103 | 54.9367 |
| | Red | 0.4588 | 0.0012 | 77.2587 | 0.0012 | 77.4904 |
| | Orange | 0.4667 | 0.0042 | 71.9572 | 0.0047 | 71.4401 |
| Object 2 | Blue | 0.3216 | 0.3851 | 52.3094 | 0.3851 | 52.3094 |
| | Green | 0.4157 | 0.2584 | 54.0419 | 0.259 | 54.0319 |
| | Yellow | 0.4471 | 0.5675 | 50.625 | 0.2834 | 53.6407 |
| | Red | 0.3529 | 0.2437 | 54.2966 | 0.2436 | 54.2986 |
| | Orange | 0.3922 | 0.3161 | 53.1662 | 0.3164 | 53,162 |

However, when using the TsTN method the background becomes neat and the object looks very clear. This is because the object color is similar to the background color when converted to grayscale as shown in Figure 3b, so that when using the Otsu method it will be difficult to distinguish but when using the TsTN method small differences can be distinguished and then corrected [8].

The majority of the TsTN method can improve image quality seen in blue, yellow, and red both on object 1 and object 2, the MSE in the TsTN method is closer to 0 than the otsu method, as well as the PSNR becomes larger. However, in the green and orange colors for both object 1 and object 2, the use of the TsTN method experienced a decrease in quality as seen from the larger MSE value and smaller PSNR. However, the difference in values is not very significant except for the yellow image, this proves that the TsTN method can help clarify and improve image segmentation when the background and object color differences are very similar.

*D.* Distance Detector Test

As is the case in the Otsu and TsTN tests which use 2 object shapes with 5 different colors (red, yellow, orange, green and blue) so that the total shapes tested are 10 objects. The distance tested on each object is 4 distances, that is 50cm, 100cm, 150cm and 200cm. The test will compare between the original distance using a meter and the distance from the camera detection results on the programmed raspberry pi. This is the results of the calculation of the distance:

TABLE 4.
DISTANCE ACCURATION TEST RESULTS

| Object | Absolute error (cm) | Relative error (%) |
|---|---|---|
| Object 1 | 0.525 | 0.539 |
| Object 2 | 0.925 | 0.831 |
| Average | 0.725 | 0.685 |

Although there is a difference between the original distance and the distance measured by the image program, the difference tends to be not too large with an overall average error of 0.725 cm or 0.685%.

The program can be used if the tendency of the object to be detected is the same because it has good accuracy.

*E. Hardware Design Results*

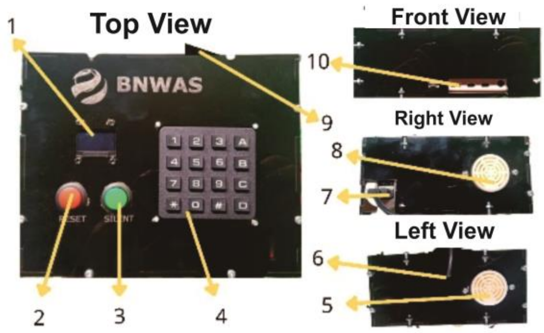


**Figure. 4.** Hardware parts

TABLE 5.
PARTS OF PROTOTYPE

| Num. | Part Name | No. | Part Name |
|---|---|---|---|
| 1 | OLED | 6 | GPS Cable |
| 2 | Reset Button | 7 | *Port* USB for Camera |
| 3 | Silent Button | 8 | *buzzer* 2 |
| 4 | *Keypad* | 9 | power switch |
| 5 | *buzzer* 1 | 10 | Raspberry Pi 4B ports |

The hardware system design in this final project has 2 main parts, namely the part that uses the ESP32 controller to control the BNWAS alarm system, and BNWAS and the part that uses the Raspberry Pi 4B to detect obstacles and their distances.

### *F. Software Design Result*

*Software* that will be used in this project is an Android-based application that will provide information related to the BNWAS alarm system, GPS, distance and connectivity, besides that the server creation process is also included in the design of this software.


**Figure. 5.** Android Display

The results of the communication test between prototype and cloud firestore server show that the average delay is 179.8 ms where if we refer to the TIPHON standard it is good while in packet loss during the test there is no data packet lost or 0%, if we refer to the TIPHON data it can be said to be very good.

### *G. System-wide Test Result*

The test results of the whole system are derived from various possible conditions. From the test results, discrepancies always occur at the GPS location, this is indeed in line with the GPS accuracy that has been tested in Table 2 where there are indeed differences between google maps and the location designation on the prototype. Another difference lies in the distance reading which has a difference, this is natural considering the accuracy of distance reading there are differences. So it can be concluded that the overall work of the prototype went well.

TABLE 6.
SYSTEM-WIDE TEST RECAPITULATION

| Test | Suitability |
|---|---|
| 1st Test | 85.71% |
| 2nd Test | 85.71% |
| 3rd Test | 85.71% |
| 4th Test | 71.42% |
| 5th Test | 71.42% |

## IV. CONCLUSION

Based on the results of testing and system analysis carried out, it can be concluded the accuracy of the alarm system system, GPS accuracy, otsu image processing and distance detection on the prototype has good accuracy with each accuracy of 1.54% on the alarm system, the accuracy error on GPS is 2,8m, respectively. then for Otsu image processing on yellow objects it improves, and others tend to be constant and the error in distance detection is 0.685%.

Hardware design with PCB design is designed using the eagle application, then for the casing design using autocad with acrylic material. And android-based software designed using Android Studio with a firebase database and for network testing has a delay of 179.8 ms and 0% packet loss. All systems that have been created have been tested and work well.

## REFERENCES

[1] Basarnas, "Data infografis kecelakaan," Basarnas, 2021. [Online]. Available: basarnas.go.id. [Accessed 26 January 2021].

[2] D. Suwardjo, J. Haluan, I. Jaya and S. Poernomo, "STUDY OF FATAL ACCIDENT RATE, ACCI-DENT PREVENTION AND MITIGATION OF FISH-ING SHIP ACCIDENTS BASED ON OPERATIONS AT PPP TEGALSARI, VAT PEKALONGAN AND PPS CILACAP," MARITEK, vol. 10, no. 1, pp. 61-72, 2010.

[3] G. Sampurno, G. Kusumawardani and I. Gunawan, "OPTIMIZATION OF USE OF BNWAS (BRIDGE NAVIGATIONAL WATCH ALARM SYSTEM) WHEN SAILING IN MV. ELINE ENTERPRISE OWNED BY PT. ADOVELIN RAHARJA," National Seminar on Maritime Interdisciplinary Studies, vol. 1, no. 1, pp. 205-211, 2019.

[4] T. V, Understanding Digital Image Processing, CRC Press, 2018.

[5] S. L. S. Abdullah, H. A. Hambali and N. Jamil, "Segmentation of Natural Images Using an Improved Thresholding-based Technique," Procedia Engineering, vol. 41, pp. 938 - 944, 2012.

[6] A. Rosebrock, "Find distance from camera to object/marker using Python and OpenCV," 2015. [Online]. Available: https://www.pyimagesearch.com/2015/01/19/find-distance-camera-objectmarker-using-python-opencv/. [Accessed 28 Januari 2021].