

RANCANG BANGUN BNWAS DILENGKAPI PERINGATAN HALANGAN BERBASIS PENGOLAHAN CITRA DAN *REAL TIME* TRACKING

Dewa Pamungkas¹, Imam Sutrisno², Afif Zuhri Arfianto³

¹²³Program Studi Teknik Otomasi, Jurusan Teknik Kelistrikan Kapal

Politeknik Perkapalan Negeri Surabaya

Jalan Teknik Kimia, Kampus ITS Sukolilo, Surabaya 60111

E-mail: dewapamungkas@student.ppns.ac.id

Abstrak

Kapal merupakan transportasi laut yang sering digunakan di Indonesia. Namun, siapa sangka transportasi tersebut justru menjadi sebuah kasus karena seringnya terjadi kecelakaan. Hal tersebut telah dibuktikan dari data KNKT (Komite Nasional Keselamatan Transportasi) yang mencatat bahwa dari tahun 2012 hingga 2017 telah terjadi peningkatan kecelakaan di perairan. Bahkan, menurut Badan Nasional Pencarian dan Pertolongan (Basarnas) pada tahun 2020 telah terjadi sebanyak 878 Kejadian dengan korban mencapai 4658 jiwa. Pada tugas akhir ini penulis membuat prototipe BNWAS yang dilengkapi dengan peringatan halangan menggunakan hasil deteksi citra dengan menggunakan metode *otsu* yang diperkuat TsTN sedangkan untuk deteksi jarak menggunakan metode *triangle similarity*, dilengkapi dengan *realtime tracking* dengan GPS dan seluruh sistem dapat diamati pada aplikasi Android berbasis *Cloud Firestore*. Tugas Akhir melakukan beberapa analisis diantaranya analisis performa dengan cara menghitung akurasi sistem alarm BNWAS, akurasi deteksi citra, akurasi jarak, akurasi GPS, akurasi pengujian sistem keseluruhan dan *packet loss*. Akurasi masing-masing sistem sangat baik karena *error* dibawah 3%, sistem dapat bekerja sebagai BNWAS dilengkapi dengan fitur pendeteksi halangan, penunjukkan lokasi beserta history lokasi dan interface android.

Kata Kunci: BNWAS, Tubrukan, Metode *Otsu*, GPS, *Triangle Similarity*.

1. PENDAHULUAN

1.1 Latar Belakang

Kapal merupakan transportasi laut yang sering digunakan di Indonesia. Namun, siapa sangka transportasi tersebut justru menjadi sebuah kasus karena seringnya terjadi kecelakaan. Hal tersebut telah dibuktikan dari data KNKT (Komite Nasional Keselamatan Transportasi) yang mencatat bahwa dari tahun 2012 hingga 2017 telah terjadi peningkatan kecelakaan di perairan. Bahkan, menurut Badan Nasional Pencarian dan Pertolongan pada tahun 2020 telah terjadi sebanyak 878 Kejadian dengan korban mencapai 4658 jiwa [1].

Kecelakaan-kecelakaan tersebut menurut data Komisi Nasional Keselamatan Transportasi mayoritas disebabkan oleh terjadinya ledakan atau kebakaran yang bersumber dari kerusakan sistem kapal, selain itu hal mengkhawatirkan lainnya kecelakaan terjadi akibat dari tubrukan dimana merupakan jumlah ketiga terbesar penyebab kecelakaan kapal tersebut. Hal ini tentu sangat mengkhawatirkan mengingat Indonesia merupakan negara maritim.

Hal yang lebih mengkhawatirkan terjadi pada nelayan kita Posisi risiko kecelakaan berada pada posisi yang *unacceptable risk* artinya perlu upaya penurunan risiko kecelakaan dalam kurun waktu satu tahun. Hal tersebut karena data yang menunjukkan angka 115 orang meninggal per 100.000 awak kapal pertahun dan masih lebih tinggi bila dibanding tingkat kecelakaan fatal kapal penangkap ikan tingkat dunia, yakni 80 orang meninggal/100.000 awak kapal. Hal tersebut disebabkan karena sistem keamanan kapal nelayan sangat kurang bahkan dapat dikatakan 70,31% tidak layak [2].

Oleh karena itu, pada karya tulis ini penulis membuat konsep sistem untuk mengurangi angka kecelakaan kapal akibat tubrukan yaitu dengan melengkapi pada sistem BNWAS yaitu ditambah dengan sistem Peringatan Halangan Berbasis Pengolahan Citra dengan metode *Otsu* dan *real time tracking* pada kapal. Prototipe BNWAS ini diharapkan juga akan membantu nelayan dalam menerapkan teknologi keamanan pada kapal khususnya untuk kapal pencari ikan 150 GT sesuai dengan standarisasi.

Sistem pada karya tulis ini akan

mengimplementasikan metode *otsu* dengan tambahan TsTN untuk mendeteksi keberadaan penghalang didepan kapal, serta *tracking* berbasis GPS dengan *IoT System*, dimana ketika terdeteksi dan adanya halangan pada kapal melalui *smartphone Android* yang berbasis *cloud firestore*. Sistem pun akan dilengkapi dengan jalur *tracking* melalui *google maps API* yang dapat diakses pada *smartphone Android* dan informasi jarak halangan.

2. METODOLOGI

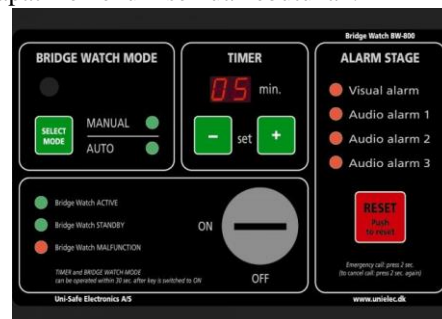
2.1 BNWAS (*Bridge Navigational Watch Alarm System*)

Bridge Navigational Watch Alarm System atau biasa disingkat dengan BNWAS mulai diperkenalkan melalui amandemen SOLAS 1974 BAB V Peraturan 19, yaitu atas persetujuan anggota IMO (*International Maritime Organization*) pada sidang *Maritime Safety Committee* yang ke 86 (MSC 86) yang dituangkan ke dalam Resolusi MSC Nomor 282 (86) pada tanggal 5 Juni 2009. SOLAS Bab V Peraturan 19 menyatakan bahwa semua kapal 150 GT dan diatas harus menginstal BNWAS. *Bridge Navigational Watch Alarm System* atau biasa disingkat dengan BNWAS, adalah peralatan yang dipasang di anjungan dan merupakan sistem alarm dinas jaga navigasi di anjungan untuk memantau aktivitas anjungan dan mendeteksi kesalahan operator yang dapat menyebabkan kecelakaan di laut. Tujuan Bridge Navigational Watch Alarm System (BNWAS) adalah untuk memantau kegiatan di anjungan dan mendeteksi kesalahan operator, yang dapat menyebabkan kecelakaan maritim, sehingga meningkatkan keselamatan navigasi dan tidak ada pengurangan pengawakan anjungan [3].

Bridge Navigational Watch Alarm System (BNWAS) akan secara otomatis aktif apabila kemudi kapal diletakkan pada posisi “autopilot” (fungsi kemudi otomatis. Persyaratan minimum BNWAS sesuai dengan ketentuan *International Maritime Organization* (IMO) adalah memiliki fungsi satu tahap diam (*dormant stage*) dan 3 tahap alarm (*alarm stage*), kecuali pada kapal-kapal penumpang dan kapal penangkap ikan, alarm tahap ke 2 boleh dihilangkan.

Meskipun pada dasarnya BNWAS adalah suatu alat yang digunakan untuk memudahkan sistem navigasi pada kapal dan dapat mencegah kelalaian dari petugas jaga (OOW).

Tapi sering kali diabaikan dan tidak digunakan karena harganya yang mahal atau kurangnya kesadaran. Khususnya untuk para nelayan yang sering mengabaikan alat-alat keselamatan minimum ini dimana yang telah memenuhi peralatan keselamatan minimal di lokasi penelitian baru mencapai 29,69%, sebanyak 70,31% belum dilengkapi peralatan keselamatan sesuai persyaratan minimal [2]. Ekonomi menjadi alasan utama dari hal ini karena alat keamanan navigasi sering kali memiliki harga mahal. Oleh karena itu sistem BNWAS yang digunakan pada karya tulis ini akan menggunakan 2 sistem alarm sesuai dengan peraturan dan fitur yang lengkap agar nelayan hanya perlu membeli satu alat dan dapat memenuhi semua kebutuhan.



Gambar 1. BNWAS

2.2 Thresholding

Thresholding adalah teknik penting dalam aplikasi segmentasi citra. Ide dasar dari thresholding adalah memilih nilai optimal ambang batas *gray-level* (*optimal gray-level threshold*) untuk memisahkan obyek yang menarik pada gambar (*object of interest*) dari latar belakangnya (*background*), berdasarkan distribusi *gray-level*nya. Karena manusia dengan mata telanjang dapat dengan mudah terdiferensiasi oleh obyek dan latar belakang yang kompleks, maka *image thresholding* merupakan tugas yang sulit untuk dilakukan, dengan tujuan memisahkan kedua hal tersebut [4].

2.3 Metode Otsu

Metode Otsu biasanya digunakan untuk menghitung nilai *threshold* secara otomatis, dengan tujuan untuk memisahkan obyek yang dimaksud (*object of interest*) dari latar belakangnya. Metode berbasis histogram ini membagi tingkat abu-abu (*gray level*) dari suatu citra menjadi dua kelas. *Gray level* tersebut terdiri dari berbagai tingkat abu-abu

yang bervariasi dari mulai hitam pada intensitas terlemah, hingga putih pada intensitas terkuat, yang berkisar dari 0 sampai 255.

Nilai *threshold* (ambang batas) kemudian digunakan untuk mengisolasi *area of interest* (daerah obyek) dari latar belakang dengan mengubah citra (gambar) *grayscale* menjadi citra biner. Citra biner terdiri dari piksel hitam dan putih, dimana piksel dengan tingkat keabuan (*gray level*) yang lebih besar dari nilai *threshold* dianggap sebagai putih dan semua piksel yang lain dianggap sebagai hitam. Citra biner yang dihasilkan adalah citra tersegmentasi, dimana “nilai 1” (putih) merepresentasikan *object of interest* dan “nilai 0” (hitam) merepresentasikan latar belakang. Namun, pada beberapa kasus bisa saja ditemukan bahwa gambar tidak memiliki bentuk yang tepat untuk merepresentasikan wilayah yang dimaksud, hal ini disebabkan karena ketidaksesuaian nilai *threshold* yang dihitung secara otomatis. Keadaan tersebut menunjukkan bahwa metode Otsu yang sudah ada, tidak dapat melakukan segmentasi pada citra natural secara benar. Oleh karena itu, modifikasi nilai-nilai *threshold* diperlukan untuk mengekstrak daerah tertentu saja (*area of interest*) [5].

2.4 TsTN

Bagian ini akan menjelaskan tentang metode TsTN, yaitu teknik segmentasi yang sudah ditingkatkan (*improved segmentation technique*) untuk citra natural. Perbaikan untuk teknik segmentasinya dilakukan dengan memodifikasi algoritma segmentasi berbasis *threshold* yang digabungkan dengan teknik invers. Metode ini disebut sebagai metode TsTN. Metode dari *improved thresholding-based segmentation technique* ini terbagi menjadi tiga langkah utama, yaitu inisialisasi nilai *threshold*, konversi citra *grayscale* menjadi citra biner, dan modifikasi nilai *threshold* yang kemudian digunakan dalam proses segmentasi citra [5].

2.5 Triangle Similarity untuk Perhitungan Jarak

Untuk menentukan jarak dari kamera penulis ke objek atau penanda yang diketahui, penulis akan menggunakan *triangle similarity*. *Triangle similarity* berjalan seperti ini: Penanda atau objek dengan lebar W yang

diketahui. Kemudian menempatkan penanda ini agak jauh D dari kamera. Gambar dari objek menggunakan kamera dan kemudian mengukur lebar jelas dalam piksel P . Ini memungkinkan kami untuk mendapatkan panjang fokus yang dirasakan F dari kamera berikut persamaannya [6].

$$F = (P \times D) / W.$$

Lalu untuk menghitung jarak kamera dengan benda dapat dihitung melalui persamaan berikut.

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

2.6 GPS

GPS merupakan navigasi yang memberikan informasi berupa posisi suatu objek di bumi, berbasis sistem satelit. Untuk dapat menentukan suatu posisi objek di bumi, GPS *receiver* membutuhkan 3 – 4 sinyal satelit yang ditangkap agar dapat membaca posisi dengan baik. Jika GPS *receiver* mampu menangkap tiga sinyal satelit, maka GPS *receiver* akan menerima data berupa *longitude* dan *latitude*.

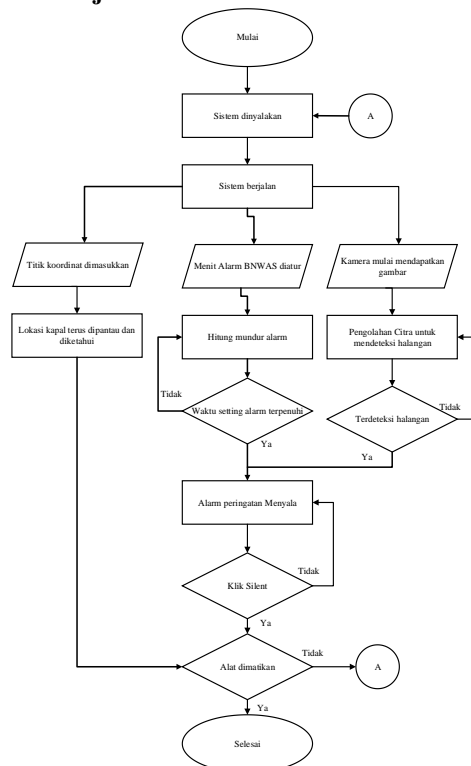


Gambar 2. GPS receiver Ublox M8N

2.7 Webcam

Webcam (singkatan dari kamera web) adalah sebutan bagi kamera waktunya yang gambarnya bisa dilihat melalui *www* (*World Wide Web*), program pengolah pesan cepat, atau aplikasi pemanggilan video. Istilah *webcam* merujuk pada teknologi secara umumnya, sehingga kata *webcam* kadang-kadang diganti dengan kata lain yang memberikan pemandangan yang ditampilkan di kamera. Kamera web dapat diartikan juga sebagai sebuah kamera video digital kecil yang dihubungkan ke komputer melalui port USB, port COM atau dengan jaringan Ethernet atau Wi-Fi.

2.8 Kerja Sistem



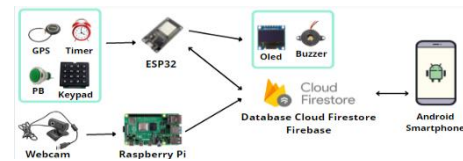
Gambar 3. Diagram alir kerja sistem

Pada **Gambar 3** menjelaskan tentang alur kerja sistem yang akan dibuat. Sistem akan diawali dengan menekan atau menyalakan alat, lalu mulai melakukan kerjanya dimana terdapat tiga blok yaitu blok GPS, blok sistem alarm dan blok kamera. Pada blok GPS sistem akan mengambil titik koordinat dan memunculkannya secara *real-time* pada aplikasi, sedangkan pada blok sistem alarm dimana akan dimulai dengan pengaturan waktu jika pengaturan waktu terpenuhi maka alarm akan berbunyi, begitupun pada kamera jika mendeteksi halangan maka akan menyalakan alarm peringatan dan peringatan akan diam ketika tombol *silent* ditekan. Serta saat kamera mendeteksi halangan jarak halangan akan dimunculkan juga.

2.9 Rancangan Alat

Gambar 4 merupakan rancangan alat yang akan dibuat oleh penulis dimana terdapat beberapa bagian utama diantaranya: Kamera yang dihubungkan dengan USB. Box 331ontroller yang berisikan raspberry pi 4, ESP32, GPS, LCD Oled, buzzer, keypad, dan tombol-tombol. Smartphone Android.

Hubungan antar *hardware* tersebut dapat dilihat pada **Gambar 4**.



Gambar 4. Hubungan Hardware

3. HASIL DAN PEMBAHASAN

3.1 Pengujian Sistem Alarm

Pada pengujian pertama ini akan diujikan sistem waktu tunda yang di program pada ESP32, lalu dibandingkan dengan *stop watch* yang ada di *smartphone android*. Pengujian ini akan mencari ralat mutlak dan ralat relatif untuk mengetahui akurasi dari sistem pemrograman dan sistem baca dari mikrokontroller pada ESP32. Pengujian akan menggunakan waktu kelipatan 2 menit, dimana waktu ini sering digunakan pada BNWAS. Berikut hasil dari pengujian tersebut.

Tabel 1 Hasil Pengujian Sistem Alarm

No.	Waktu Program (detik)	Waktu <i>Stop watch</i> (detik)	Ralat mutlak (detik)	Ralat Relatif (%)
1.	120	121	1	0,82
		121	1	0,82
		122	2	1,65
2.	240	244	4	1,63
		245	5	2,04
		244	4	1,63
3.	360	366	6	1,63
		365	5	1,36
		367	7	1,90
4.	480	487	7	1,43
		489	9	1,84
		487	7	1,43
5.	600	610	10	1,63
		610	10	1,63
		610	10	1,63
6.	720	732	12	1,63
		732	12	1,63
		731	11	1,50
Total <i>error</i>				1,54

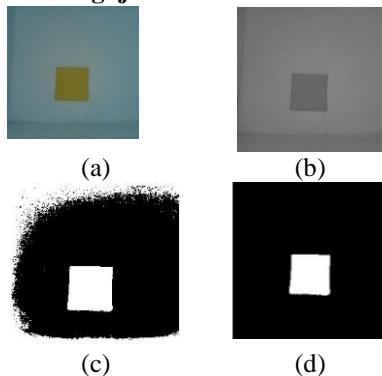
3.2 Pengujian Sistem GPS

Pengujian kedua ini akan mengujikan akurasi dari sistem pemrograman GPS pada ESP32 dengan menggunakan *hardware* GPS Ublox Neo M8N. **Gambar 5** merupakan rangkaian pengujian yang digunakan untuk pengujian. Pengujian akan mengambil data *latitude* dan *longitude* kemudian dibandingkan dengan yang ada pada *google maps* di *smartphone android*. Pengujian akan dilakukan di 10 tempat yang berbeda, nantinya akan dibandingkan dan menghasilkan selisih. Berikut merupakan hasil dari pengujiannya.

Tabel 2 Hasil Pengujian GPS

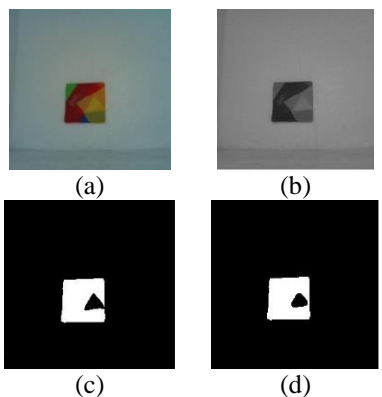
No.	Pada Google Maps		Pada Prototipe		Jarak (m)
	Latitude	Longitude	Latitude	Longitude	
1.	-6.823490	107.639505	-6.823481	107.639510	1,14
2.	-6.846577	107.598730	-6.846595	107.598709	3,06
3.	-6.818412	107.624260	-6.818392	107.624285	3,54
4.	-6.817637	107.622842	-6.817661	107.622862	3,46
5.	-6.863967	107.594595	-6.863987	107.594582	2,64
6.	-6.838504	107.598281	-6.838519	107.598299	2,59
7.	-6.843537	107.598583	-6.843549	107.598597	2,04
8.	-6.863594	107.594413	-6.863574	107.594426	2,64
9.	-6.858265	107.595176	-6.858288	107.595194	3,23
10.	-6.868626	107.593605	-6.868601	107.593627	3,69
Rata-rata					2,8

3.3 Hasil Pengujian Perubahan Citra



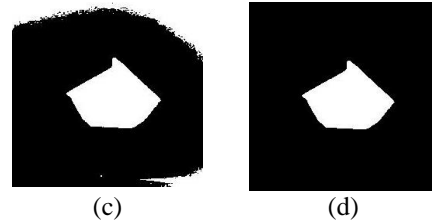
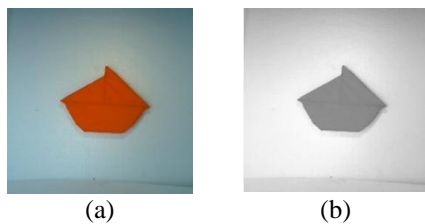
Gambar 5. Perubahan Citra Objek Warna Kuning

(a) Citra asli (b) Citra greyscale (c) Citra otsu (d) Citra TsTN



Gambar 6. Perubahan Citra Objek Warna Acak

(a) Citra asli (b) Citra greyscale (c) Citra otsu (d) Citra TsTN



Gambar 7. Perubahan Citra Objek Bentuk Acak

(a) Citra asli (b) Citra greyscale (c) Citra otsu (d) Citra TsTN

Berdasarkan perbandingan gambar secara kasat mata, citra *output* menggunakan metode otsu dan TsTN seperti yang terlihat pada gambar tidak terlihat perbedaan yang signifikan kecuali pada citra berwarna kuning seperti pada Gambar 5. Pada Gambar 5 pada metode otsu bagian *background* terlihat acak dan kurang rapi walaupun objek tetap terlihat namun jika dibiarkan maka akan bermasalah atau menyebabkan *error* ketika digunakan untuk pengenalan objeknya nanti. Tetapi, ketika menggunakan metode TsTN *background* menjadi rapi dan objek terlihat sangat jelas. Hal ini disebabkan oleh warna objek mirip dengan warna *background* ketika di ubah menjadi *greyscale* seperti yang terlihat pada Gambar 5b, sehingga ketika menggunakan metode otsu akan sulit dibedakan tetapi ketika menggunakan metode TsTN perbedaan kecil dapat dibedakan kemudian diperbaiki.

Tabel 3 Hasil Pengujian MSE dan PSNR

No.	Nama Objek	Warna	Threshold (T)	Otsu		TsTN	
				MSE	PSNR	MSE	PSNR
1.	Objek 1	Biru	0,4569	0,000576	80,5606	0,000512	81,0721
2.		Hijau	0,4392	0,1057	57,9242	0,1059	57,915
3.		Kuning	0,4667	0,2673	53,8951	0,2103	54,9367
4.		Merah	0,4588	0,0012	77,2587	0,0012	77,4904
5.		Jingga	0,4667	0,0042	71,9572	0,0047	71,4401
6.		Acak	0,4627	0,0025	74,137	0,0026	74,0284
7.	Objek 2	Biru	0,3216	0,3851	52,3094	0,3851	52,3094
8.		Hijau	0,4157	0,2584	54,0419	0,259	54,0319
9.		Kuning	0,4471	0,5675	50,625	0,2834	53,6407
10.		Merah	0,3529	0,2437	54,2966	0,2436	54,2986
11.		Jingga	0,3922	0,3161	53,1662	0,3164	53,162
12.		Acak	0,3686	0,3445	52,7934	0,3437	52,8031
13.	Objek Acak	Biru	0,5373	0,001	77,9945	0,0013	76,9328
14.		Jingga	0,7529	0,2447	54,279	0,1019	58,0849
15.		Hijau	0,7098	0,5804	50,5275	0,1097	57,7629
16.		Merah	0,5529	0,0013	77,0927	0,0015	76,5332
17.		Kuning	0,4392	0,1621	56,0679	0,7004	49,711
18.		Acak	0,3961	0,0608	60,3269	0,0607	60,3338

Mayoritas metode TsTN dapat memperbaiki kualitas citra terlihat pada warna biru, jingga, dan merah baik pada objek 1 dan objek 2, MSE pada metode TsTN lebih mendekati 0 dibandingkan metode otsu, begitupun dengan PSNRnya menjadi lebih besar. Tetapi pada warna hijau dan jingga baik pada objek 1 dan objek 2, penggunaan metode

TsTN mengalami penurunan kualitas terlihat dari nilai MSE yang lebih besar dan PSNR yang lebih kecil. Namun perbedaan nilai memang tidak terlalu signifikan kecuali pada citra berwarna kuning, hal ini membuktikan bahwa metode TsTN dapat membantu memperjelas dan memperbaiki segmentasi citra ketika perbedaan warna latar dan objek sangat mirip, walau dapat juga sebaliknya terlihat pada objek acak dimana segmentasi malah memperjelek citra.

3.4 Pengujian Pendeteksi jarak

Seperti halnya pada pengujian otsu dan TsTN dimana menggunakan 3 bentuk objek dengan 6 warna yang berbeda (merah, kuning, hiau, biru, jingga dan acak) sehingga total bentuk yang diujikan adalah 10 objek. Jarak yang diujikan pada setiap objek yaitu 4 jarak yaitu 50cm, 100cm, 150cm dan 200cm. Pengujian akan membandingkan antara jarak asli menggunakan meteran dan juga jarak hasil deteksi kamera pada *raspberry pi* yang telah diprogram. Berikut merupakan hasil pengujian perhitungan dari jarak:

Tabel 4 Hasil pengujian jarak Statis

Objek	Ralat mutlak (cm)	Ralat relatif (%)
Objek 1	0,52	0,54
Objek 2	0,93	0,83
Objek Acak	8,44	7,63
Rata-rata	3,3	3

Meskipun terdapat perbedaan antara jarak asli dan jarak hasil pengukuran program citra, tetapi perbedaan cenderung tidak terlalu besar rata-rata *error* keseluruhan 3,3 cm atau 3%. Program dapat digunakan jika kecenderungan objek yang akan terdeteksi sama karena memiliki akurasi yang baik.

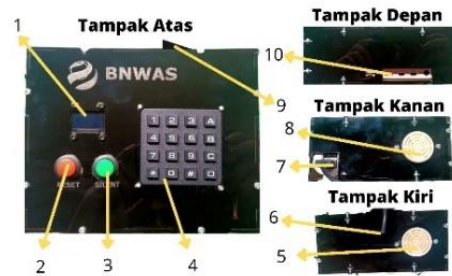
Tabel 5 Hasil pengujian jarak Dinamis

Objek	Ralat mutlak (cm)	Ralat relatif (%)
Objek 1	1,2	1
Objek 2	1,15	1,06
Objek Acak	1,18	1,11

Rata-rata	1,19	1,11
-----------	------	------

Tabel 5 merupakan hasil pengujian dinamis masing-masing perwakilan dari setiap objek. Dari hasil pengujian rata-rata *error* yang teridentifikasi adalah 1,06%. Hasil ini membuktikan bahwa program yang digunakan memiliki performa yang baik walaupun objek yang dideteksi bergerak.

3.5 Hasil Perancangan Hardware



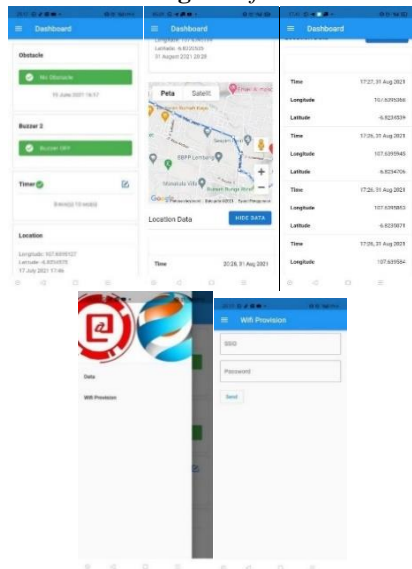
Gambar 8. Bagian hardware

Tabel 6 Bagian-bagian luar prototipe

No .	Nama Bagian	No .	Nama Bagian
1	OLED	6	Kabel GPS
2	Tombol Reset	7	Port USB untuk Kamera
3	Tombol Silent	8	Buzzer 2
4	Keypad	9	Saklar power
5	Buzzer 1	10	Ports Raspberry Pi 4B

Perancangan sistem *hardware* pada karya tulis ini terdapat 2 bagian utama yaitu bagian yang menggunakan kontroller ESP32 untuk mengendalikan sistem alarm BNWAS, dan BNWAS serta bagian yang menggunakan Raspberry Pi 4B untuk mendeteksi halangan beserta jaraknya

3.6 Hasil Perancangan Software



Gambar 9. Tampilan android software

Software yang akan digunakan pada karya tulisan kali ini adalah aplikasi berbasis Android yang akan memberikan informasi terkait sistem alarm BNWAS, GPS, jarak dan konektivitas, selain itu proses pembuatan server juga termasuk pada perancangan software ini. Proses pembuatan software menggunakan aplikasi Android Studio dengan menggunakan bahasa pemrograman Kotlin.

Hasil pengujian komunikasi antara prototipe dan server cloud firesotre memperlihatkan bahwa rata-rata delay yaitu dari ESP32 ke server 764,2ms dengan maksimal jarak dengan router sejauh 10m dan server ke android 716,57ms, lalu untuk lost pada ESP32 ke server sebanyak 1,4 dengan maksimal jarak dengan router sejauh 10m dan server menuju android 1.

3.7 Hasil Pengujian seluruh sistem

Tabel 7. Rekapitulasi pengujian sistem

Pengujian Ke-	Kesesuaian
Pengujian 1	85,71%
Pengujian 2	85,71%
Pengujian 3	85,71%
Pengujian 4	71,42%
Pengujian 5	71,42%

Hasil pengujian keseluruhan sistem yang berasal dari berbagai kondisi yang memungkinkan. Dari hasil pengujian tersebut ketidak sesuaian selalu terjadi pada lokasi GPS hal ini memang sejalan juga dengan akurasi GPS yang telah diujikan pada Tabel 7 dimana memang terdapat perbedaan antara google maps dan penunjukkan lokasi pada prototipe. Perbedaan lain terletak pada pembacaan jarak dimana memiliki perbedaan, hal tersebut wajar

terjadi mengingat akurasi pembacaan jarak memang terdapat perbedaan. Sehingga dapat disimpulkan bahwa keseluruhan kerja dari prototipe berjalan dengan baik.

4. PENUTUP

4.1 Kesimpulan

Berdasarkan hasil yang telah didapatkan pada analisa dan pengujian sistem yang telah dibuat, maka dapat disimpulkan bahwa:

1. Tingkat akurasi sistem alarm system, akurasi GPS, pengolahan citra otsu dan deteksi jarak pada prototype memiliki akurasi yang baik dengan masing-masing akurasi yaitu 1,54% pada alarm system, error akurasi pada GPS adalah 2,8m, lalu untuk pengolahan citra otsu yang diperkuat TsTN pada umumnya citra membaik dan tersegmentasi namun ada beberapa citra yang tidak terdeteksi karena saat proses greyscale warna terdeteksi menjadi sama dengan background, dan yang lainnya cenderung konstan, serta error pada pendeteksian jarak untuk statis bernilai 3% sedangkan untuk dinamis 1,06%.
2. Perancangan hardware dengan desain PCB didesain menggunakan aplikasi eagle, lalu untuk desain casing menggunakan autocad dengan bahan akrilik. Serta software berbasis android yang didesain dengan menggunakan Android Studio dengan database firebase. Sistem dapat bekerja sebagai BNWAS dilengkapi dengan fitur pendeteksi halangan, penunjukkan lokasi beserta history lokasi dan interface android.
3. Seluruh sistem yang telah dibuat telah diujikan dan memiliki kerja yang baik.

4.2 Saran

Berdasarkan percobaan yang telah dilakukan terdapat beberapa saran untuk melanjutkan dan memperbaiki kekurangan yang ada pada Tugas Akhir ini, yaitu :

1. Pada pendeteksian jarak dengan objek halangan, metode yang digunakan sebaiknya menggunakan stereo cam agar dapat mendeteksi kedalaman, sehingga objek yang dideteksi bisa objek dengan bentuk tak beraturan.
2. Jika kondisi memungkinkan pengujian sangat diperlukan pada kapal yang akan diterapkan.
3. Fitur pada aplikasi android dapat ditambahkan dengan fish finder untuk

kapal penangkap ikan dan kapal penumpang atau cargo dapat ditambahkan fitur penunjukkan jalur.

5. DAFTAR PUSTAKA

- Basarnas, "Data info grafis kecelakaan," Basarnas, 2021. [Online]. Available: basarnas.go.id. [Accessed 26 01 2021].
- D. Suwardjo, J. Haluan, I. Jaya and S. H. Poernomo, "Kajian Tingkat Kecelakaan Fatal, Pencegahan Dan Mitigasi Kecelakaan Kapal-Kapal Penangkap Ikan Yang Berbasis Operasi Di Ppp Tegalsari, Ppn Pekalongan Dan Pps Cilacap," *MARITEK*, vol. 10, no. 1, pp. 61-72, 2010.
- G. P. Sampurno, G. Kusumawardani and I. Gunawan, "Optimalisasi Penggunaan Bnwas (Bridge Navigational Watch Alarm System) Pada Saat Berlayar Di Mv. Eline Enterprise Milik Perusahaan Pt. Adovelin Raharja," *National Seminar on Maritime Interdisciplinary Studies*, vol. 1, no. 1, pp. 205-211, 2019.
- H. J. Vala and A. Baxi, "Review on Otsu Image Segmentation Algorithm," *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, vol. 2, pp. 387-389, 2013.
- 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.
- R. A, "Intersection over union (iou) for object detection," 2016. [Online]. Available: <https://www.pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/>. [Accessed 15 January 2021].