



Calhoun: The NPS Institutional Archive
DSpace Repository

Theses and Dissertations

1. Thesis and Dissertation Collection, all items

2017-09

Development of a vision-based situational awareness capability for unmanned surface vessels

Toh, Ying Jie Benjemin

Monterey, California: Naval Postgraduate School

<http://hdl.handle.net/10945/56185>

Copyright is reserved by the copyright owner.

Downloaded from NPS Archive: Calhoun



Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community. Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

<http://www.nps.edu/library>



**NAVAL
POSTGRADUATE
SCHOOL**

MONTEREY, CALIFORNIA

THESIS

**DEVELOPMENT OF A VISION-BASED SITUATIONAL
AWARENESS CAPABILITY FOR UNMANNED
SURFACE VESSELS**

by

Ying Jie Benjamin Toh

September 2017

Thesis Advisor:
Second Reader:

Oleg Yakimenko
Fotis Papoulias

Approved for public release. Distribution is unlimited.

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			<i>Form Approved OMB No. 0704-0188</i>	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington, DC 20503.				
1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE September 2017	3. REPORT TYPE AND DATES COVERED Master's thesis		
4. TITLE AND SUBTITLE DEVELOPMENT OF A VISION-BASED SITUATIONAL AWARENESS CAPABILITY FOR UNMANNED SURFACE VESSELS			5. FUNDING NUMBERS	
6. AUTHOR(S) Ying Jie Benjemin Toh				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			10. SPONSORING / MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB number ___N/A___.				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release. Distribution is unlimited.			12b. DISTRIBUTION CODE	
13. ABSTRACT (maximum 200 words) The current generations of unmanned surface vessels (USVs) are reliant on the human operator for collision avoidance. This reliance poses a constraint on the operational envelope of the USV as it requires a high bandwidth and low latency communication link between the USV and control station. This thesis adopts a systems engineering approach in identifying the capability gap and the factors that drive the need for a USV with autonomous capability. An algorithm employing edge detection and morphological structuring methods is developed in this thesis to explore the feasibility of using a computer vision-based technique to provide a situational awareness capability, which is required to achieve autonomous navigation. The algorithm was tested with both color video imagery and infrared video imagery, and the results obtained from processing the images demonstrated the viability of using this information to provide situational awareness to the USV. It is recommended that further work be done to improve the robustness of the algorithm.				
14. SUBJECT TERMS Unmanned systems, situational awareness			15. NUMBER OF PAGES 81	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UU	

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release. Distribution is unlimited.

**DEVELOPMENT OF A VISION-BASED SITUATIONAL AWARENESS
CAPABILITY FOR UNMANNED SURFACE VESSELS**

Ying Jie Benjamin Toh
Civilian, Singapore Technologies Electronics Limited
B.Eng (Hons), National University of Singapore, 2007

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN SYSTEMS ENGINEERING

from the

**NAVAL POSTGRADUATE SCHOOL
September 2017**

Approved by: Oleg Yakimenko
Thesis Advisor

Fotis Papoulias
Second Reader

Ronald Giachetti
Chair, Department of Systems Engineering Department

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The current generations of unmanned surface vessels (USVs) are reliant on the human operator for collision avoidance. This reliance poses a constraint on the operational envelope of the USV as it requires a high bandwidth and low latency communication link between the USV and control station. This thesis adopts a systems engineering approach in identifying the capability gap and the factors that drive the need for a USV with autonomous capability. An algorithm employing edge detection and morphological structuring methods is developed in this thesis to explore the feasibility of using a computer vision-based technique to provide a situational awareness capability, which is required to achieve autonomous navigation. The algorithm was tested with both color video imagery and infrared video imagery, and the results obtained from processing the images demonstrated the viability of using this information to provide situational awareness to the USV. It is recommended that further work be done to improve the robustness of the algorithm.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I.	INTRODUCTION.....	1
	A. BACKGROUND	1
	B. CHALLENGES OF UNMANNED SYSTEMS.....	1
	C. MOTIVATION OF STUDY	2
	D. PROBLEM FORMULATION AND RESEARCH QUESTIONS	2
	E. THESIS ORGANIZATION.....	3
II.	LITERATURE REVIEW	5
	A. AUTOMATIC SYSTEMS VS. AUTONOMOUS SYSTEMS	5
	B. LEVELS OF AUTONOMY	6
	C. MAKING SENSE OF THE ENVIRONMENT.....	7
	D. SITUATION AWARENESS.....	7
	E. LEVELS OF SITUATION AWARENESS	8
	F. ELEMENT OF TIME IN SITUATION AWARENESS	9
	G. CLASSES OF USV	9
III.	SYSTEMS ENGINEERING APPROACH	15
	A. CAPABILITY DEFICIENCY	15
	B. NEEDS ANALYSIS.....	16
	C. FUNCTIONAL DECOMPOSITION.....	17
	D. FUNCTIONAL FLOW	18
IV.	COMPUTER VISION–BASED TECHNIQUE FOR MOTION ESTIMATION.....	19
	A. GENERAL IDEA.....	19
	B. IMAGERY PREPROCESSING.....	20
	C. SHIP LOCALIZATION.....	22
	D. SHIP CHARACTERIZATION	26
V.	PROCESSING ELECTRO-OPTICS AND INFRARED IMAGERY	29
	A. EXPERIMENTS USING EO IMAGERY	29
	B. EXPERIMENTS USING INFRARED VIDEO	34
VI.	CONCLUSION AND RECOMMENDATIONS.....	51
	A. CONCLUSION	51
	B. RECOMMENDATIONS.....	52

APPENDIX. SHAPE PROPERTIES	53
LIST OF REFERENCES.....	57
INITIAL DISTRIBUTION LIST	59

LIST OF FIGURES

Figure 1.	Republic of Singapore Navy’s MCM USV Undergoing Testing. Source: Wong (2017).....	6
Figure 2.	X-Class USV. Source: Department of the Navy (2007).	10
Figure 3.	Harbor Class USV. Source: Department of the Navy (2007).....	11
Figure 4.	Snorkeler Class USV. Source: Department of the Navy (2007).....	12
Figure 5.	Fleet Class USV. Source: Department of the Navy (2007).	13
Figure 6.	Functional Decomposition for Autonomous Navigation.....	17
Figure 7.	Top-Level Functional Flow Block Diagram.....	18
Figure 8.	Functional Flow for Comprehend Environment.....	18
Figure 9.	Image of Model Representing a Ship.....	19
Figure 10.	Bounding Box Ratio Plot for Model.....	20
Figure 11.	Image Extracted from Video.....	20
Figure 12.	Original Grayscale Image before Removing Background.....	21
Figure 13.	Original Grayscale Image with Background Removed	21
Figure 14.	Contrast Adjusted Image.....	22
Figure 15.	Image after Applying Edge Detection Method.....	23
Figure 16.	Image Output from MATLAB after Dilating the Image	24
Figure 17.	Image Output from MATLAB after Fill Operation.....	24
Figure 18.	Effects of Removing Objects Connected to the Border.....	25
Figure 19.	Convex Hull Image.....	25
Figure 20.	Example of an Image with Bounding Box Overlay	27
Figure 21.	Ratio of Bounding Box against Relative Orientation of Ship.....	30
Figure 22.	Subsequence of Images of Ship Turning (180°–270°).....	30

Figure 23.	Subsequence of Images of Ship Turning (270°–360°).....	31
Figure 24.	Second Bounding Box Ratio at One Frame per Second.....	32
Figure 25.	Binary Image after Applying the Mask in YCbCr Color Space.....	33
Figure 26.	Binary Image after Applying Mask in HSV Color Space.....	34
Figure 27.	Image Extracted from IR_Video1.....	35
Figure 28.	Intensity Plot of Background-Removed Image.....	36
Figure 29.	Output from YCbCr Masking.....	37
Figure 30.	Image after Applying Contrast Adjustment.....	37
Figure 31.	IR Image Processing: Dilation (a), Filling the Holes (b), Clearing Borders (c), Creating a Convex Hull (d).....	38
Figure 32.	Bounding Box Ratio Plot for IR_Video1.....	39
Figure 33.	Frame 19 of IR_Video1.....	39
Figure 34.	Images 1 (a), 50 (b), 150 (c), and 250 (d) from IR_Video2.....	40
Figure 35.	Output from YCbCr Masking of IR_Video2 Image.....	41
Figure 36.	IR_Video2 Image after Contrast Adjustment.....	41
Figure 37.	Output from Each Step of Image Processing Routine: Dilation (a), Filling the Holes (b), Clearing Borders (c), Creating a Convex Hull (d).....	42
Figure 38.	Bounding Box Ratio Plot for IR_Video2.....	43
Figure 39.	Output from Each Step of Image Processing Routine for Frame Number 107: Dilation (a), Filling the Holes (b), Clearing Border (c), Creating a Convex Hull (d).....	43
Figure 40.	Bounding Box Ratio Plot for IR_Video1 without Contrast Enhancement.....	44
Figure 41.	Frame 2 from IR_Video1.....	45
Figure 42.	Frame 4 from IR_Video1.....	45
Figure 43.	Frame 19 of IR_Video1 without Contrast Enhancement.....	46

Figure 44.	IR_Video2 Bounding Box Ratio Plot without Contrast Enhancement.....	47
Figure 45.	IR_Video2 Frame 97 Bounding Box Image	47
Figure 46.	IR_Video2 Frames 33 (a) and 94 (b) with Bounding Box Superimposed.....	48
Figure 47.	Plot of Bounding Box Height against Video Frames.....	49
Figure 48.	Comparison of Bounding Box Height over Consecutive Frames 113 (a) and 114 (b).....	49

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

Table 1.	Four Levels of Autonomy. Source: Department of Defense (2011).....	6
Table 2.	Image Object Measurements.....	26

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

ASW	anti-submarine warfare
DOD	Department of Defense
EW	electronic warfare
EO	electro-optics
HSV	hue, saturation, and value
IR	infrared
MIO	maritime interdiction operations
RSN	Republic of Singapore Navy
MCM	mine countermeasures
RGB	red-green-blue
USV	unmanned surface vehicle
SA	situational awareness
SOF	special operations forces
SUW	surface warfare
YCbCr	luminance-chrominance

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

Using unmanned surface vessels (USVs) for “dull, dirty and dangerous missions” is gaining traction in recent years as it removes the human from a potentially life-threatening environment in missions such as mine hunting or maritime interdiction (Department of Defense 2011, 17). Current USVs rely on human operators sitting in remote control stations, either on land or onboard ships, to monitor the vessels’ surroundings and to perform collision detection and avoidance. This reliance on the human operator constrains the operating envelope of the USV as it requires a high bandwidth and low latency communication link for safe operations, especially in waters with heavy traffic.

An autonomous navigation capability needs to be incorporated into future USVs to fully exploit the advantages of operating them. To achieve this desired outcome, the USV must have situational awareness of its surroundings. This thesis adopts a systems engineering approach for identifying the capability gap in today’s USV and the factors that drive the need for a USV with autonomous navigation capability. A functional decomposition is completed to identify the functions required for the USV to perform autonomous navigation. This thesis uses a computer vision-based technique to implement one of the functions identified through the functional decomposition.

The algorithm, developed in MATLAB, converts the video into individual frames before enhancing them for further processing. The images undergo processing using edge detection and morphological structuring techniques before information is derived from the processed images. The algorithm was tested with images from color video sources as well as infrared (IR) video sources. One of the key challenges encountered during this process was that shadows caused the information derived from the images to be inaccurate. While developing the algorithm, several methods were tested with different parameters to determine the most effective method for removing background noise from the images. It was found that filtering using the mean intensity value in the image was effective with color video images, but it did not work with the IR video images; instead,

filtering the IR video images in the luminance-chrominance color space was found more effective.

Information was derived by analyzing the bounding box that was drawn by the algorithm around the objects detected in the images. The boat's orientation could be inferred by comparing the bounding box ratio over time. Similarly, by comparing the height of the bounding box over time, information such as whether the boat is sailing away or toward the camera could be inferred.

This thesis demonstrated the feasibility of using a computer vision-based technique to provide a situational awareness capability to a USV. Future work focusing on removing the shadows in the images is recommended to improve the robustness of the algorithm and reliability of the information derived from the images. Another area of possible research is fusing the information derived from the algorithm with data from other sensors onboard the USV to improve the situational awareness capability.

References

Department of Defense. 2011. *Unmanned Systems Integrated Roadmap: FY2011–2036*. Reference No. 11-S-3613. Washington, DC: Department of Defense.

ACKNOWLEDGMENTS

I would like to thank Professor Oleg Yakimenko for his guidance and advice throughout the whole research period, as well as Associate Professor Fotis Papoulias for his advice that contributed to this thesis. I would also like to express my appreciation to Senior Lecturer Barbara Berlitz for reviewing this thesis and helping refine it.

I would also like to thank Pete Arslanian and David Devon from NAVAIR, Flight Controls 4.3.2.6, for providing the infrared video used in the experiments.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. BACKGROUND

Unmanned surface vessel (USV) use in military operations is not something new. USVs have been used as early as the period after World War II to conduct minesweeping operations and to test the radioactivity of water after atomic bomb tests (Department of Defense 2011). They were also used during the Vietnam War to perform minesweeping operations.

In recent years, the Republic of Singapore Navy has used the “Protector” USV in anti-piracy operations in the Gulf of Aden for surveillance and force protection missions (Republic of Singapore Navy 2017). These operations are deemed “dull, dirty or dangerous” for humans, which is why the USV is best suited. The use of USVs in such operations removes the human from a potentially life-threatening environment, thus reducing the probability of human casualties. Apart from the aforementioned missions, there are also plans to use USVs in anti-submarine, surface, and electronic warfare as well as for support missions using special operations forces and maritime interdiction operations (Department of the Navy 2007).

B. CHALLENGES OF UNMANNED SYSTEMS

With increasing use of USVs in a range of missions, there are several challenges to overcome, as identified in the Department of Defense’s *Unmanned Systems Integrated Roadmap FY2011–2036*, if the full potential of the unmanned systems is to be realized. One of the consequences of the expanding roles of unmanned systems in current operations is the burden of the additional manpower required to operate these systems while simultaneously operating manned systems. The DOD identifies “autonomy” as having the potential to reduce the manpower requirement in the operations of unmanned systems because multiple unmanned systems may fall under the control of a single operator. The other benefit to increasing the level of autonomy of the unmanned systems is that high communication bandwidth is no longer a prerequisite. Currently, many USVs can self-navigate by following a set of waypoints or a planned path. However, there is

still reliance on the remote operator to monitor the video sent back from the USV for potential obstacles and to intervene to avoid collision with other ships in the vicinity (Roberts and Sutton 2006). In order for the operator to intervene in a timely manner, the video must have a certain level of fidelity and low latency, which severely limit the operational range of the USV. Satellite communications do not support such stringent requirements imposed on the communication channel. An autonomous USV would perform collision avoidance on its own without operator inputs; hence, there would be no need to transmit high-quality real-time video from the USV to the remote operator to monitor the surroundings and to perform collision avoidance maneuvers.

C. MOTIVATION OF STUDY

One of the requirements for a USV to navigate autonomously is the ability to have situational awareness (SA) of the environment in which it is operating. Current USVs rely mainly on radar to provide SA of its surroundings. Although the USV is equipped with a navigation camera, it is mainly used for the remote operator to monitor the surroundings. This study explores using an image processing algorithm to analyze the video images from the navigation camera to provide another level of SA capability for the USV.

D. PROBLEM FORMULATION AND RESEARCH QUESTIONS

The USV relies on sensors, such as radar, electro-optics (EO), and an infrared (IR) camera, to obtain SA of its surroundings. Depending on the size of the USV, it may not have the capacity to carry all these sensors. Current USVs, which rely on man-in-the-loop operations, would at a minimum have an EO camera for the human operator to monitor the USV surroundings. Therefore, the problem addressed in this thesis is whether a computer vision-based technique can be used to provide an SA capability for USVs.

This thesis addresses the following research questions:

- (1) Can a computer vision-based technique be used with EO imagery to provide situational awareness for the USV?
- (2) Can a computer vision-based technique be used with IR imagery to provide situational awareness for the USV?

- (3) How do environmental factors affect the computer vision–based technique?

E. THESIS ORGANIZATION

To address the aforementioned research questions, this thesis is organized as follows. Chapter II presents the literature review, providing a topic overview and defining autonomy and situational awareness. It also provides information on the categorizations of USVs currently being developed. Chapter III describes the systems engineering approach used to explore the research questions. The current deficiency in capability is identified, and a needs analysis is performed to guide development of the solution. Chapter IV presents the algorithm this research developed to address the problem and the challenges faced during its development. The chapter describes the series of steps taken to derive information from the video images and discusses on the results. Chapter V summarizes the development of the algorithm and its results. This chapter also proposes recommendations for future work.

THIS PAGE INTENTIONALLY LEFT BLANK

II. LITERATURE REVIEW

A. AUTOMATIC SYSTEMS VS. AUTONOMOUS SYSTEMS

Automatic systems are fully preprogrammed and act repeatedly and independent of external influence or control. An automatic system can be described as self-steering or self-regulating and is able to follow an externally given path while compensating for small deviations caused by external disturbances. However, the automatic system is not able to define the path according to some given goal or to choose the goal dictating its path. (Department of Defense 2011, 43)

A ship's autopilot can be viewed as an automatic system that keeps the ship sailing at a predetermined speed and bearing. The autopilot compensates for the resistance caused by waves by controlling the throttle on the ship to maintain a preset speed. Another example of an automatic system is the cruise control used in some motor vehicles to maintain a constant speed defined by the driver.

In *Unmanned Systems Integrated Roadmap FY2011–2036*, the Department of Defense (DOD) explains that an autonomous system “is self-directed by choosing the behavior it follows to reach a human-directed goal” (Department of Defense 2011, 43). For example, an unmanned surface vehicle (USV) with mine countermeasures (MCM) and an autonomous capability will be able to plan its own transit path and scanning pattern based on the area, which is defined by the operator. Therefore, the autonomous system's ability to make decisions based on a set of rules or strategies for achieving a human-directed goal is its key difference from an automatic system. A prototype of the Republic of Singapore Navy's MCM USV undergoing sea trials in Singapore is shown in Figure 1.



Figure 1. Republic of Singapore Navy’s MCM USV Undergoing Testing.
Source: Wong (2017).

B. LEVELS OF AUTONOMY

The four levels of autonomy as defined by the DOD in *Unmanned Systems Integrated Roadmap FY2011-2036* are shown in Table 1.

Table 1. Four Levels of Autonomy. Source: Department of Defense (2011).

Level	Name	Description
1	Human Operated	A human operator makes all decisions. The system has no autonomous control of its environment although it may have information-only responses to sensed data.
2	Human Delegated	The vehicle can perform many functions independently of human control when delegated to do so. This level encompasses automatic controls, engine controls, and other low-level automation that must be activated or deactivated by human input and must act in mutual exclusion of human operation.
3	Human Supervised	The system can perform a wide variety of activities when given top-level permissions or direction by a human. Both the human and the system can initiate behaviors based on sensed data, but the system can do so only if within the scope of its currently directed tasks.
4	Fully Autonomous	The system receives goals from humans and translates them into tasks to be performed without human interaction. A human could still enter the loop in an emergency or change the goals, although in practice there may be significant time delays before human intervention occurs.

C. MAKING SENSE OF THE ENVIRONMENT

In order for a USV to operate autonomously in a complex and uncertain environment, such as the in the busy Singapore Strait or the Gulf of Aden, it must have situational awareness of the environment in which it is operating. The autonomous system must have the capability to sense the environment through its different onboard sensors and to convert all their data into useful information to make decisions as to the course of action for achieving its goal (Department of Defense 2011).

D. SITUATION AWARENESS

Situational awareness (SA) is about obtaining information of what is in the environment that is related to the tasks or goals that a person is trying to achieve. In addition, the ability to comprehend this information is important as it helps in the decision-making process in the courses of actions to achieve a particular goal. It is also the ability to use this information to predict future events that can aid in deciding future courses of actions to achieve the particular goal. According to Endsley and Jones (2011),

SA is being aware of what is happening around you and understanding what that information means to you now and in the future. This awareness is usually defined in terms of what information is important for a particular job or goal. The concept of SA is usually applied to operational situations, where people must have SA for a specified reason, for example, in order to drive a car, treat a patient, or separate traffic as an air traffic controller. Therefore, SA is normally defined as it relates to the goals and objectives of a specific job or function. Only those pieces of the situation that are relevant to the task at hand are important for SA. (13)

The formal definition of SA is as follows: “The perception of the elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future” (Endsley and Jones 2011, 13)

The operational scenario assumed for this research is of an USV navigating autonomously in an area where there are other vessels and ships sailing. In order to navigate autonomously, the USV needs to know what other vessels are around it and where these vessels are heading, so it does not collide with them.

E. LEVELS OF SITUATION AWARENESS

There are three levels of SA that derive from the formal definition. They are as follows:

- Level 1 – Perception of the elements in the environment
- Level 2 – Comprehension of the current situation
- Level 3 – Projection of future status

(1) Level 1 – Perception of Elements in the Environment

In *Designing for Situation Awareness*, Endsley and Jones (2011) explain, “The first step in achieving SA is to perceive the status, attributes, and dynamics of relevant elements in the environment” (14). A USV uses its onboard sensors to sense its environment; these sensors may include a radar, navigation camera, or thermal imager. The relevant elements in this case would be the other vessels sailing in the vicinity and other obstacles such as navigation buoys or land masses.

(2) Level 2 – Comprehension of the Current Situation

The next level of SA as defined by Endsley and Jones (2011) is “understanding what the data and cues perceived mean in relation to relevant goals and objectives” (16). In the context of navigating autonomously in a busy strait with many other ships nearby, the USV must be able to integrate the data received from multiple sensors to form information that is relevant to its task of performing autonomous navigation.

(3) Level 3 – Projection of Future Status

According to Endsley and Jones (2011), “Once the person knows what the elements are and what they mean in relation to the current goal, it is the ability to predict what those elements will do in the future (at least in the short term) that constitutes Level 3 SA” (18). Within the context of navigating autonomously in a busy strait, a USV with this level of SA must be capable of determining which of the detected obstacles may become collision threats.

F. ELEMENT OF TIME IN SITUATION AWARENESS

The timeliness of the information gathered plays an important role in SA. For example, a ship captain needs to know well ahead of time whether there are any obstacles ahead to steer the ship to avoid collision. If this piece of information is not relayed to the captain in a timely manner, any action taken subsequently may be insufficient to prevent a collision. The other aspect of time in SA is that the course of action depends on the amount of time available before an event occurs. In the ship example, if there is sufficient time to react from the point the obstacle is detected, the ship could slow down and change heading to avoid it. However, if there is insufficient time for the ship to slow down, the captain may have to take drastic measures by commanding the ship to go full astern, executing an emergency stop to minimize damage to the ship.

G. CLASSES OF USV

In *The Navy Unmanned Surface Vehicle (USV) Masterplan*, the Department of the Navy (2007) establishes four classes of USVs based on mission requirements and the characteristics of the vessel such as stability, payload fraction, tow power, and endurance. The four classes of USV derived from the analysis are presented in the following paragraphs.

(1) X-Class (Small)

The X-Class USV, as shown in Figure 2, is a small special purpose craft measuring three meters or shorter. Its main purpose is to support the mission needs of special operations forces or maritime interdiction operations. This class of USV has limited endurance, payload, and seakeeping ability (Department of the Navy 2007, 59).



Figure 2. X-Class USV. Source: Department of the Navy (2007).

(2) Harbor Class (7m)

The Harbor Class USV, as shown in Figure 3, is based on a seven-meter rigid hull inflatable boat with moderate endurance. Its main role is to perform intelligence, surveillance, and reconnaissance as well as maritime security missions (Department of the Navy 2007, 60).

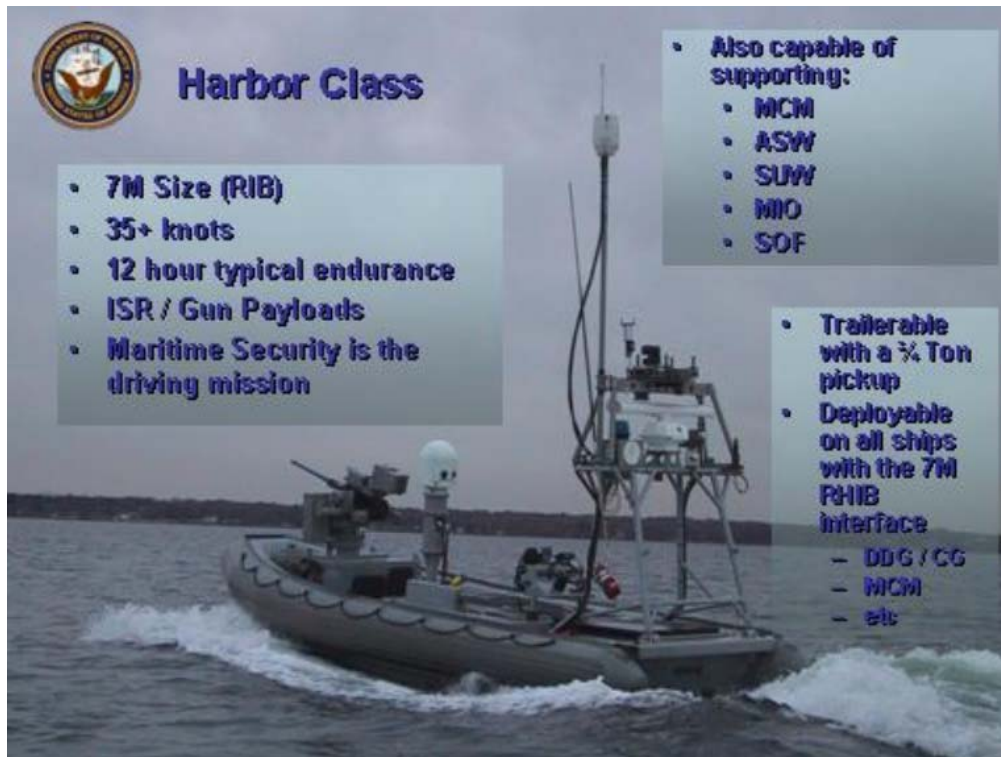


Figure 3. Harbor Class USV. Source: Department of the Navy (2007).

(3) Snorkeler Class (Semi-submersible)

The Snorkeler Class USV, as shown in Figure 4, is a seven-meter semi-submersible craft designed mainly for MCM search and neutralization missions as well as anti-submarine (ASW) missions. The craft is submerged during operations, which gives it an advantage compared to other surface hull types in high-sea states, as it is much more stable (Department of the Navy 2007, 61).



Figure 4. Snorkeler Class USV. Source: Department of the Navy (2007).

(4) Fleet Class (11m)

The Fleet Class USV is an 11-meter planing or semi-planing hull craft, as shown in Figure 5. It has moderate speed and endurance while towing payloads for MCM missions. It can also be deployed for ASW, surface warfare, or electronic warfare missions as it operates at high speed and has very long endurance (Department of the Navy 2007, 62).



Figure 5. Fleet Class USV. Source: Department of the Navy (2007).

This chapter presented a literature review on the definition of autonomy and situation awareness. The next chapter describes the systems engineering approach adopted in the development of the algorithm.

THIS PAGE INTENTIONALLY LEFT BLANK

III. SYSTEMS ENGINEERING APPROACH

Attaining the goal of an autonomous capability in USVs requires an interdisciplinary approach as it involves teams with differing specializations such as sensors, image processing, as well as navigation and control. A systems engineering process is best suited to develop this capability to address the problem formulated in Chapter I.

A. CAPABILITY DEFICIENCY

The first step in the systems engineering process is to identify the problem before proceeding to define the need. A problem exists when there is a gap between the desired state and the current state.

(1) Current State

Current USVs still rely heavily on human operators to control them and to monitor their surroundings to identify and avoid potential collision threats. Hence, high bandwidth communication links are required to provide high fidelity video for operators to perform these tasks effectively. In addition, the communication link must be without high latency, which results in delays in the information presented to the operators and untimely actions that could lead to collisions.

(2) Desired State

The vision for future USVs is that they will be able to carry out their missions based on the goals defined by the operators. The USVs perform the tasks to accomplish the goal without human intervention. The USV is able to adapt to the changes in the operating environment based on rules or strategies defined by the human operator.

(3) Gap in Current Capability

Current USVs lack the levels of autonomy required to operate without a man in the loop for live feedback and control. USVs with an autonomous capability must have some form of situational awareness for them to make decisions without human operators.

B. NEEDS ANALYSIS

There are several factors that drive the need to develop USVs with autonomous navigation capability.

(1) Communication Links

Navigation video takes up a large portion of the data transmitted from the USV back to the remote control station. It also requires a low latency link for the operator to make timely decisions in maneuvering to avoid collisions. The issue of latency is even more critical when the USV is transiting at high speed. A USV that is capable of performing autonomous navigation eliminates the requirement of a human operator who constantly monitors the USV's surroundings to identify potential collision threats. Hence, there is no need to transmit high fidelity video back from the USV—which requires high bandwidth.

(2) Manpower

In *Unmanned Systems Integrated Roadmap FY2011-2036*, the Department of Defense (DOD) highlights, “Today’s unmanned systems require significant human interaction to operate. As these systems . . . are fielded in greater numbers, the demand for manpower will continue to grow. The appropriate application of autonomy is a key element in reducing this burden” (Department of Defense 2011, 44). The manpower issue is even more acute for nations with ageing populations such as Singapore. According to Kor Kian Beng’s article in the *Straits Times*, the Singapore Armed Forces are set to face a one-third reduction in manpower supply (Kor 2017).

(3) Susceptibility to Communications Jamming

The USV could be deployed in hostile territory where communications may be jammed by an adversary. Hence, a USV that requires constant communication with the remote control station for its operation will be rendered ineffective. However, a USV that has an autonomous capability will be able to operate in a communications-denied environment because it does not rely on a human located in the remote control station.

(4) Cooperative or Collaborative Coordination among Multiple Vehicles

Operating a group of USVs to perform large-scale missions, such as anti-submarine or mine countermeasures missions, will not be feasible without an autonomous capability. The amount of communication bandwidth required will be too great to carry out an effective mission if every USV has to be tele-operated from a remote control station. Building autonomy into the USVs means they can operate in a cooperative or collaborative manner without human intervention.

C. FUNCTIONAL DECOMPOSITION

A functional decomposition methodology is used to determine the functions required for the USV to perform autonomous navigation. The functional decomposition for achieving autonomous navigation is shown in Figure 6.

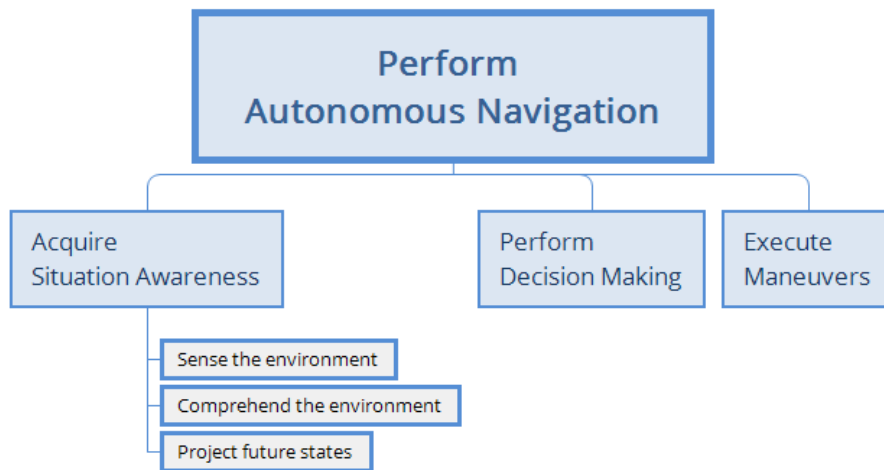


Figure 6. Functional Decomposition for Autonomous Navigation

The top-level functions are to acquire situation awareness, perform decision making, and execute maneuvers. The function to acquire situation awareness can be further decomposed into sensing and comprehending the environment as well as projecting future states.

D. FUNCTIONAL FLOW

The top-level functional flow block diagram is shown in Figure 7. In this thesis, the focus is on the comprehend environment function. This function processes the sensor data from onboard the USV; in this case, the information is a video from a navigation camera.

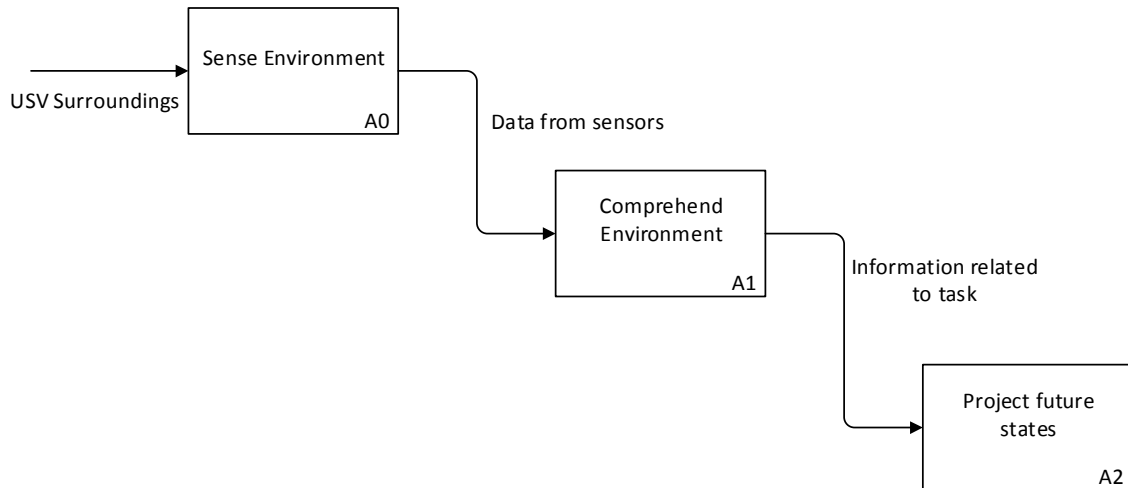


Figure 7. Top-Level Functional Flow Block Diagram

The functions within the comprehend environment block can be further decomposed into the functional flow block diagram shown in Figure 8.

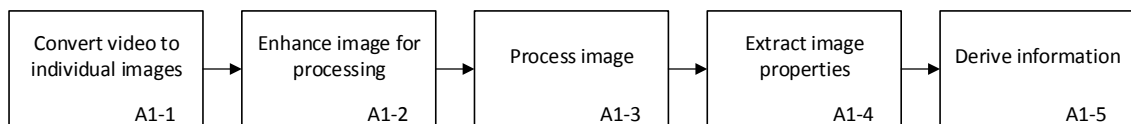


Figure 8. Functional Flow for Comprehend Environment

This chapter summarized the SE approach used to identify the capability deficiency and analyze the needs for bridging the capability gap. A functional decomposition and a functional flow were used to identify the top-level functions that the algorithm must execute.

IV. COMPUTER VISION–BASED TECHNIQUE FOR MOTION ESTIMATION

This chapter describes the steps in the computer vision–based technique developed to provide situational awareness for the unmanned surface vehicle (USV). The concept of this technique is to use a ship’s characteristics from imagery to determine its orientation. The technique consists of the following: preliminary steps to enhance the images, an algorithm to localize the ship, and measurements to characterize the ship.

A. GENERAL IDEA

A model was constructed in the MATLAB environment to test the algorithm without any external influences such as background objects or shadows. The model was constructed to turn from 180 degrees (the ship’s bow facing the camera) to 360 degrees (the ship’s stern facing the camera), which is similar to the movement of the ship in the electro-optics (EO) video imagery. An image of the model representing the ship with a bearing of 270 degrees is shown in Figure 9.

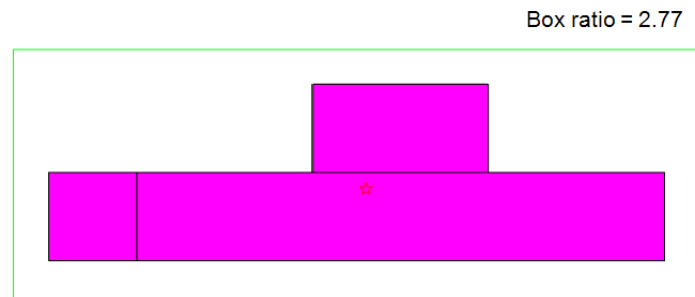


Figure 9. Image of Model Representing a Ship

The bounding box ratio plotted against the ship’s orientation is shown in Figure 10. The plot shows that when the ship’s bow or stern is facing the camera, the ratio is at its minimum value. The value is at its maximum when the port or starboard side of the ship is facing the camera. This model demonstrates the feasibility of deducing the orientation of the boat from the bounding box ratio obtained after processing the images.

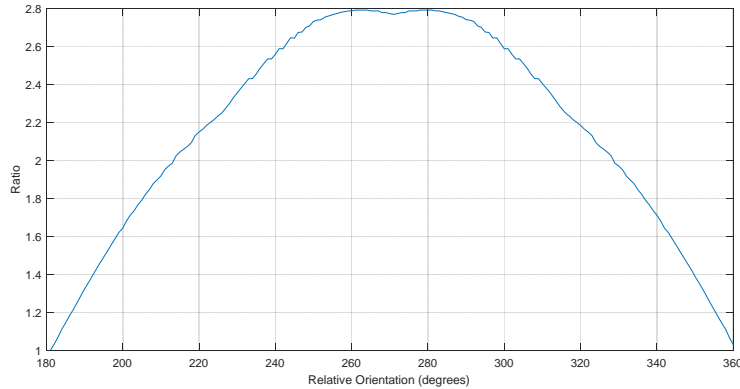


Figure 10. Bounding Box Ratio Plot for Model

All black and white images shown in this chapter have their colors inverted to reduce the amount of ink used when the thesis is printed.

B. IMAGERY PREPROCESSING

The image processing algorithm was developed in the MATLAB development environment. The algorithm follows the functional flow described in Chapter III, Section D.

The first step in the algorithm routine is to extract the individual frames from the video and convert them into images before processing. The frame rate of the video is sixty frames per second; however, from the experiments performed, it was sufficient to extract the frames at one-third the video's frame rate without losing fidelity in the information. The individually extracted frames are in red-green-blue (RGB) format as shown in Figure 11.



Figure 11. Image Extracted from Video

Each RGB image is converted to grayscale format before its background is removed. The intensity plot of the original grayscale image is shown before the background is removed (Figure 12) and afterward (Figure 13).

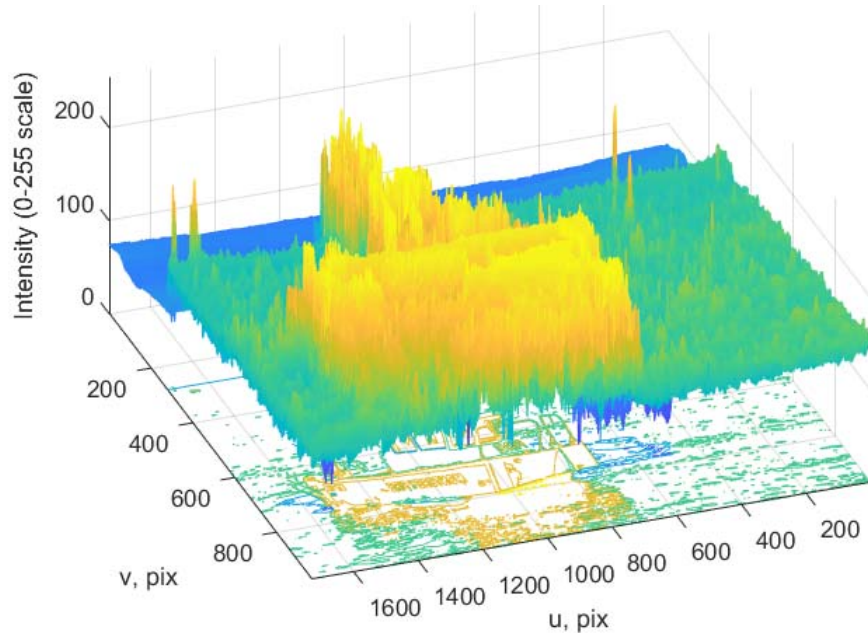


Figure 12. Original Grayscale Image before Removing Background

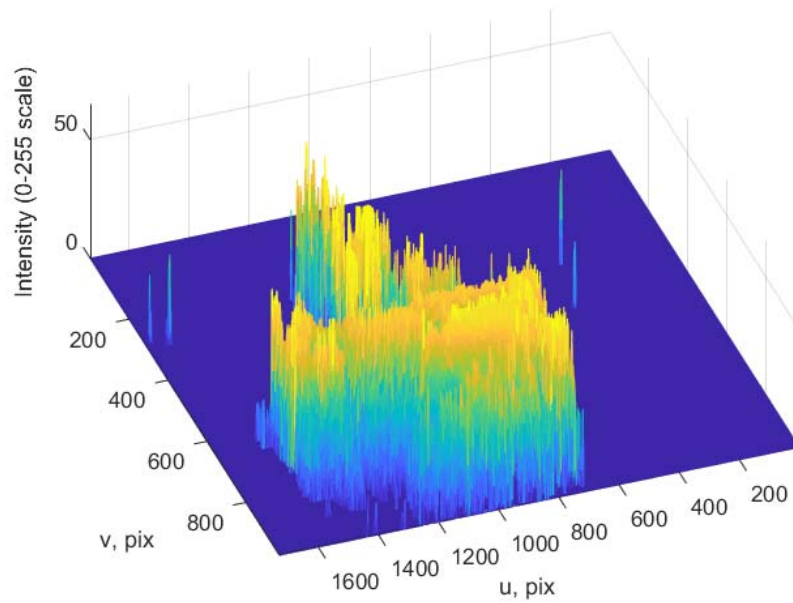


Figure 13. Original Grayscale Image with Background Removed

After the image's background is removed, the contrast of the image is enhanced using bottom-hat filtering. This step allows for more accurate edge detection in the subsequent stage without including background objects. The bottom-hat filtered image is shown in Figure 14.



Figure 14. Contrast Adjusted Image

C. SHIP LOCALIZATION

The following steps are taken to allow for image property characterization at a later stage. According to Mathworks Edge detection is an image processing technique used for finding the boundaries of objects within images. It works by detecting discontinuities in brightness. Edge detection is used for image segmentation and data extraction in areas such as image processing, computer vision, and machine vision. (Mathworks 2017a)

There are several edge detection methods in MATLAB. They are as follows:

- (1) Sobel
- (2) Prewitt
- (3) Roberts
- (4) Log
- (5) Zerocross
- (6) Canny

The Sobel edge detection method uses the Sobel Operator to detect the edges in the image. The Sobel operator, also known as the Sobel-Feldman operator, was first developed in 1968 by Irwin Sobel and Gary Feldman at the Stanford Artificial Intelligence Project. The idea was presented in a seminar at that time as “A 3x3 Isotropic Gradient Operator for Image Processing.” The Sobel operator applies a pair of 3x3 convolution masks on the image vertically and horizontally to measure the gradient in each direction. The absolute magnitude is then obtained from the summation of both the gradients in vertical and horizontal directions (Sobel 1990).

In MATLAB, a threshold is established from the gradients of the pixels in the image; pixels for which the gradient magnitude is greater than the threshold are treated as edges. The output of the image after applying edge detection is shown in Figure 15.

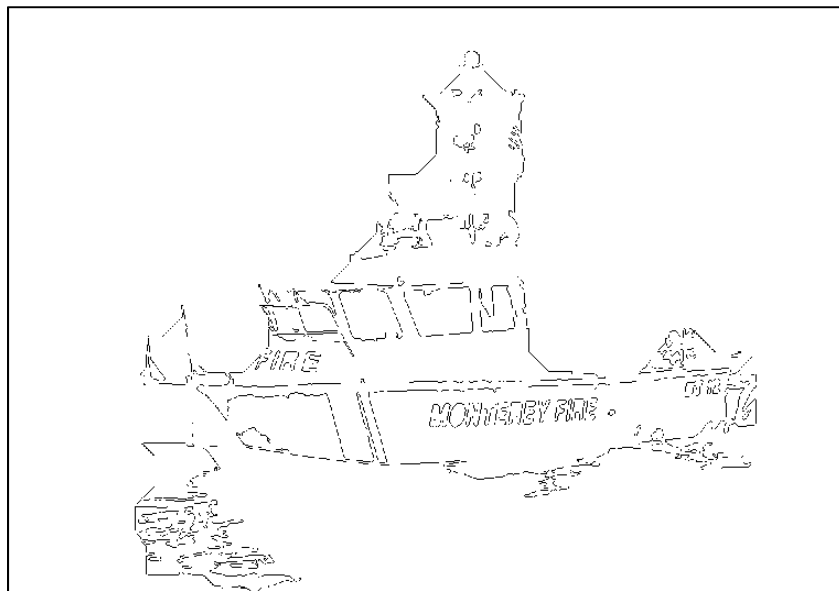


Figure 15. Image after Applying Edge Detection Method

The next step is to dilate the image to eliminate edge discontinuities. Dilation adds pixels to the boundaries of objects in an image based on the size and structuring elements (Mathworks 2017c). The structuring elements used in this algorithm are an octagon and a diagonal line. The output of the image after dilation is shown in Figure 16. This step is used to connect the edges that were detected in the previous step.

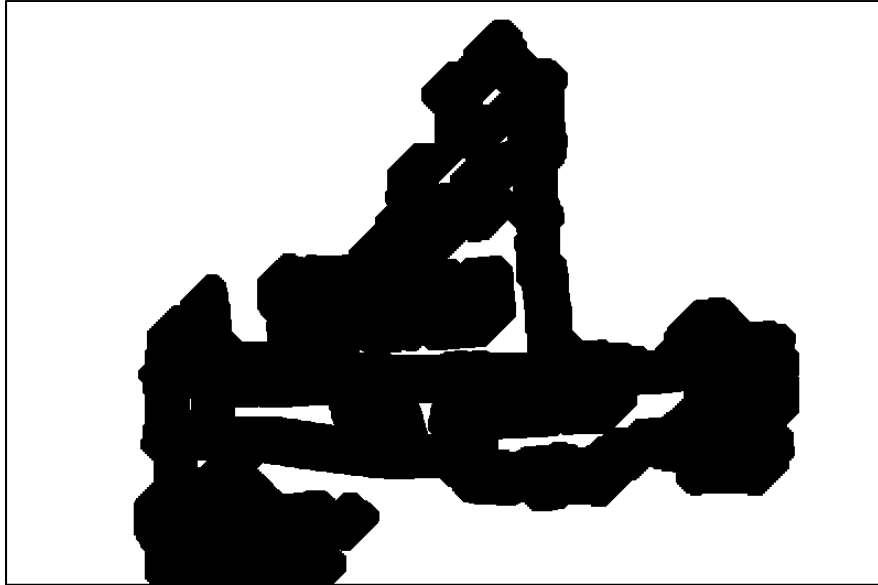


Figure 16. Image Output from MATLAB after Dilating the Image

Next, the holes in the image are filled, as shown in Figure 17. A hole is defined as a set of background pixels that cannot be captured by filling in the background from the edge of the image (Mathworks 2017b).



Figure 17. Image Output from MATLAB after Fill Operation

The next step involves removing pixels that are connected to the border of the image. Presumably, these pixels include noise and the target of interest in the center of the image. The image is shown before removing the pixels connected to the border (Figure 18a) and afterward (Figure 18b).

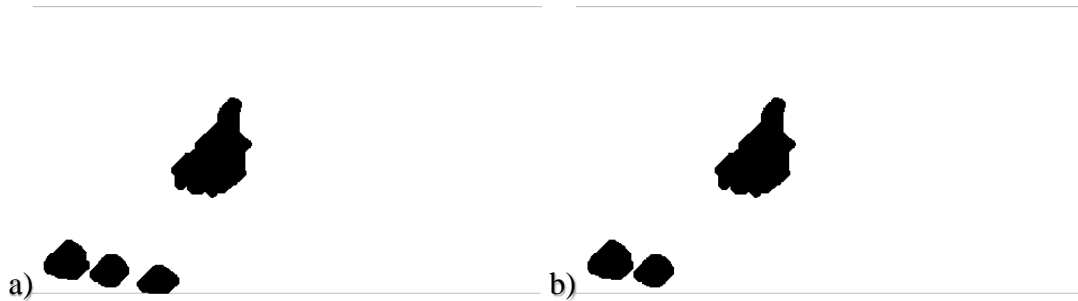


Figure 18. Effects of Removing Objects Connected to the Border

The final step in processing the image is to generate a convex hull image to aid in the subsequent extraction of image properties. The object's edges in the image are smoothed, as shown in Figure 19.

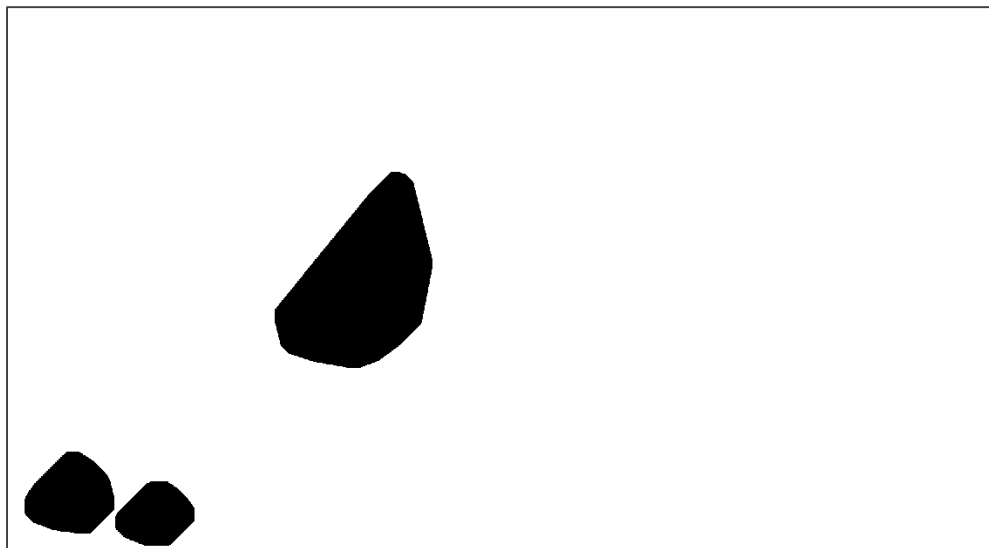


Figure 19. Convex Hull Image

D. SHIP CHARACTERIZATION

The next step is to measure and analyze the objects in the image to derive information from it. The properties that can be measured in MATLAB using the “regionprops” function are described in the shape properties table in the Appendix.

The regionprops function returns the measurements for the properties found in the Appendix as a structural array for each object in the image. The measurements for all the objects found in the image are shown in Table 2. The area property represents the number of pixels of each object in the image. The centroid property specifies the center of mass of each object; it is represented by the x-coordinate followed by the y-coordinate. The bounding box property is represented by the upper left corner coordinates of the bounding box and the dimensions of the bounding box.

Table 2. Image Object Measurements

Area (px²)	Centroid (px)	Bounding Box (px)
15766	[119,865]	[39,789,155,140]
11040	[265,898]	[195,840,137,111]
57785	[621,500]	[469,308,272,337]

Most likely, the ship will occupy the greatest number of pixels in the image. Hence, to filter out the other objects, the object with the maximum area is found based on the area property. The index for this object is then used to extract the information for the bounding box and Centroid, which correspond to the object with the largest area.

The bounding box and Centroid data are used to overlay a bounding box on the original image for visual verification that the correct object is selected, as shown in Figure 20.



Figure 20. Example of an Image with Bounding Box Overlay

This chapter presented the steps in the algorithm to process the images, with their corresponding outputs shown after applying each operation. The next chapter presents the results from testing the algorithm with EO and infrared imagery.

THIS PAGE INTENTIONALLY LEFT BLANK

V. PROCESSING ELECTRO-OPTICS AND INFRARED IMAGERY

This chapter describes the results and the challenges faced while testing the algorithm with electro-optics and infrared (IR) imagery. The different methods attempted during the development of the algorithm to remove background noise are also described in this chapter.

A. EXPERIMENTS USING EO IMAGERY

The bounding box's aspect ratio is used to infer the orientation of the ship. The aspect ratio of the bounding box is found by dividing the width of the box by the height of the box.

The bounding box ratio plotted against the ship's orientation is shown in Figure 21. There were a total of 90 images in this series, each frame representing one-third of a second of the video.

The plot begins with the bow of the ship facing the camera and then turning clockwise 180 degrees. A subset of the sequence of images showing the movement of the ship as it turns is shown in Figure 22 and Figure 23. As shown in Figure 21, when the ship's bow or stern faces the camera, the ratio is at its minimum value. The ratio starts to increase as the port side of the ship turns toward the camera.

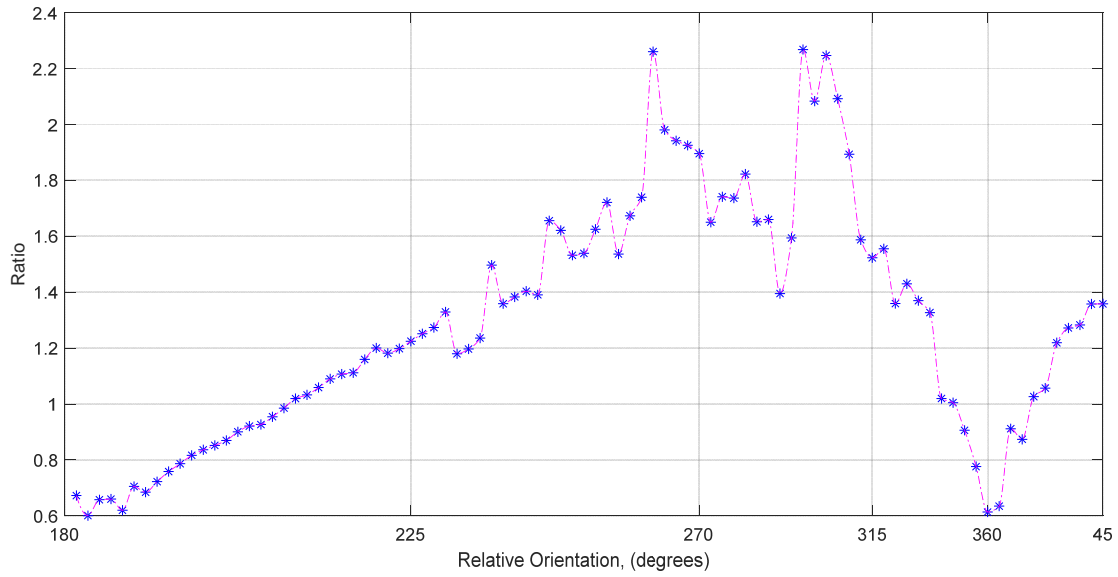


Figure 21. Ratio of Bounding Box against Relative Orientation of Ship



Figure 22. Subsequence of Images of Ship Turning (180°–270°)



Figure 23. Subsequence of Images of Ship Turning (270° – 360°)

Another experiment was conducted to investigate the effects of sampling the video at a lower frequency. There were a total of 30 frames in this experiment; instead of having three frames representing each second of the video, each second of the video was represented by one frame. The results of the bounding box ratio is shown in Figure 24. As shown, the plot with 30 frames exhibits similar trends to the one with 90 frames. Both plots show that the ratio is at a minimum when the ship's bow or stern faces the camera. As the ship starts to turn starboard, the ratio increases until the ship's port side faces the camera. The ratio decreases as it continues to turn with the stern facing the camera.

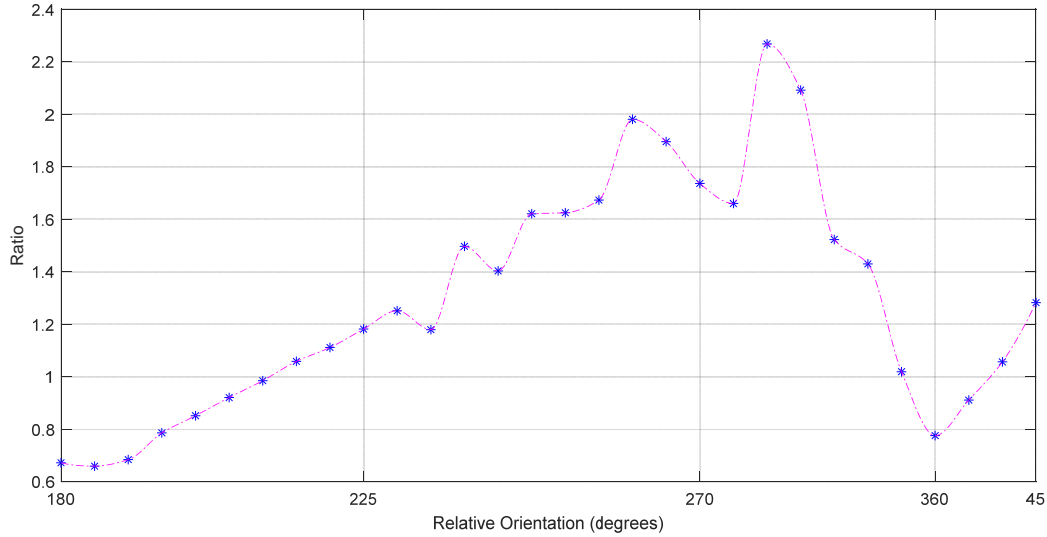


Figure 24. Second Bounding Box Ratio at One Frame per Second

One of the challenges processing the images was the effect of shadows therein. The shadows caused the image processing algorithm to mistakenly treat the shadows as part of the ship. This skewed the ship’s dimensions, thereby affecting the calculation of the bounding box ratio.

One method for removing shadows from images is to convert the images into a different color space format to find properties that are unique to the shadows. The first attempt to address the shadow problem involved converting the red-green-blue (RGB) format image into the luminance-chrominance (YCbCr) format. In the YCbCr format, the luminance information is stored in the Y component while the chrominance information is stored as two color-difference components, Cb and Cr. The mask is constructed by modifying the thresholds for the luminance channel. The results of applying the mask are shown in Figure 25. The shadows share similar properties with parts of the ship, as shown by the white background of the image. Therefore, if shadows are removed, a large portion of the ship will also be removed.

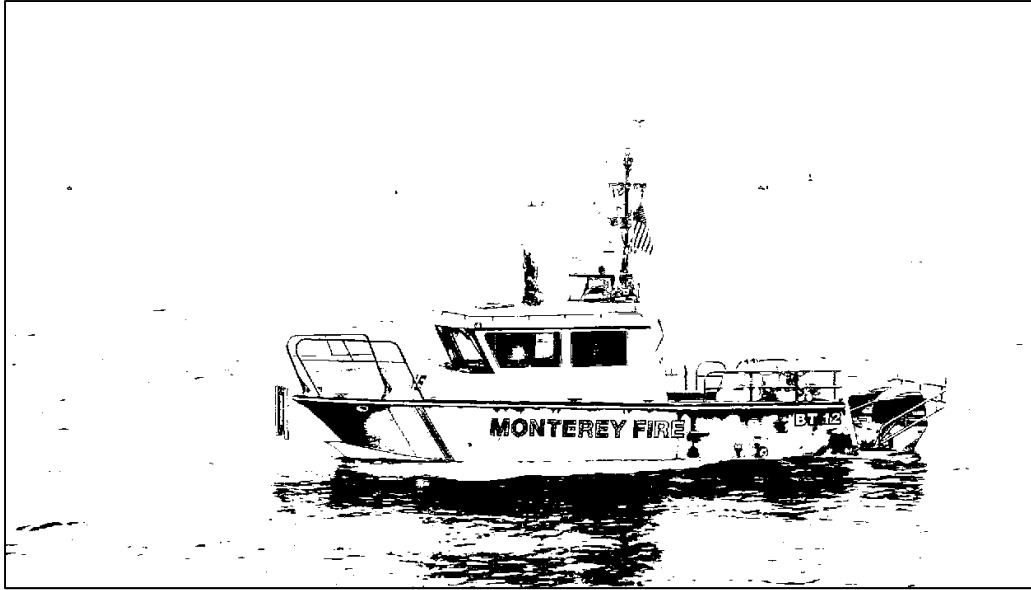


Figure 25. Binary Image after Applying the Mask in YCbCr Color Space

The second attempt converted the image into hue, saturation, and value (HSV) color space. The hue of the image represents the color of the image. As the hue increases from zero to one, it corresponds with color changes from red through green, cyan, blue, magenta, and back to red. The saturation of the image represents how much white is in the color. A value of one represents one of the hue colors, for example pure red. The value parameter represents the brightness of the color. The masked image is constructed by modifying the saturation and value thresholds. The results after applying the mask are shown in the binary image in Figure 26.

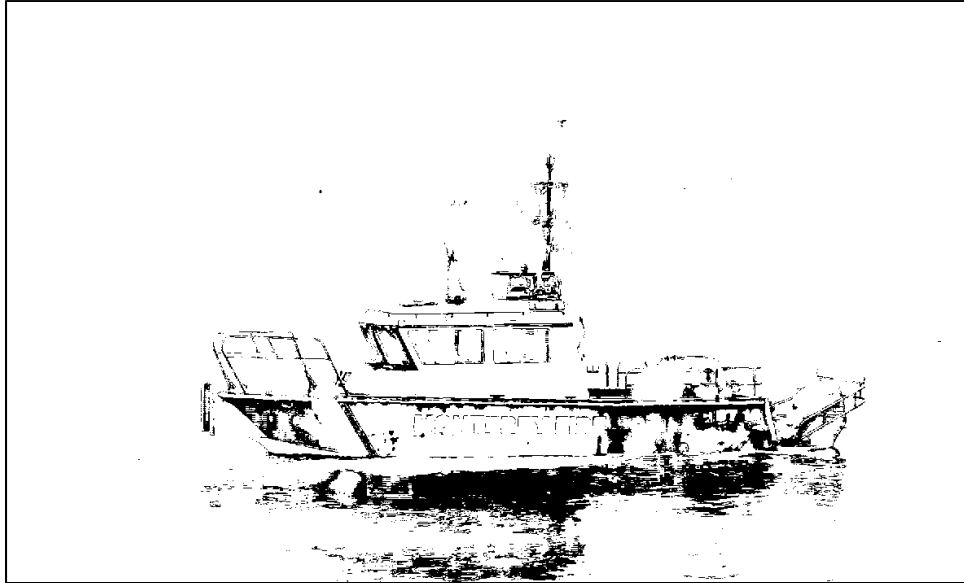


Figure 26. Binary Image after Applying Mask in HSV Color Space

Due to the shadows in the image sharing similar characteristics with the ship, either in terms of luminance or saturation, it was not possible to filter out the shadow of the ship in the water from the images using the aforementioned techniques.

B. EXPERIMENTS USING INFRARED VIDEO

Experiments were carried out to test the algorithm using IR video to see whether the problems of shadows can be overcome. Experiments were conducted with two different videos. In the first, the ship's starboard side faces the camera, moving from left to right (hereafter known as IR_Video1). In the second, the ship moves away from the camera, making slight adjustments to its path, with the stern facing the camera (hereafter known as IR_Video2).

(1) Processing IR_Video1

One of the video frames that was extracted from IR_Video1 is shown in Figure 27. Each video frame was extracted in RGB format although to the naked eye the image is only in black and white.



Figure 27. Image Extracted from IR_Video1

As shown in Figure 27, there are several noise sources in the image. The shoreline can be seen on the top edge of the video, the timestamp of the video is on the top left hand corner of the image, dark spots representing birds can be seen in the foreground, and the shadow of the ship shows in the water. To reduce the errors in the calculation of the bounding box ratio, it is necessary to remove these instances of background noise from the image before further processing.

The algorithm developed for processing video in color requires modification to process the IR video images. The same routine used to remove the background cannot be used as it will result in an image with all the pixels having the same intensity, as shown in Figure 28.

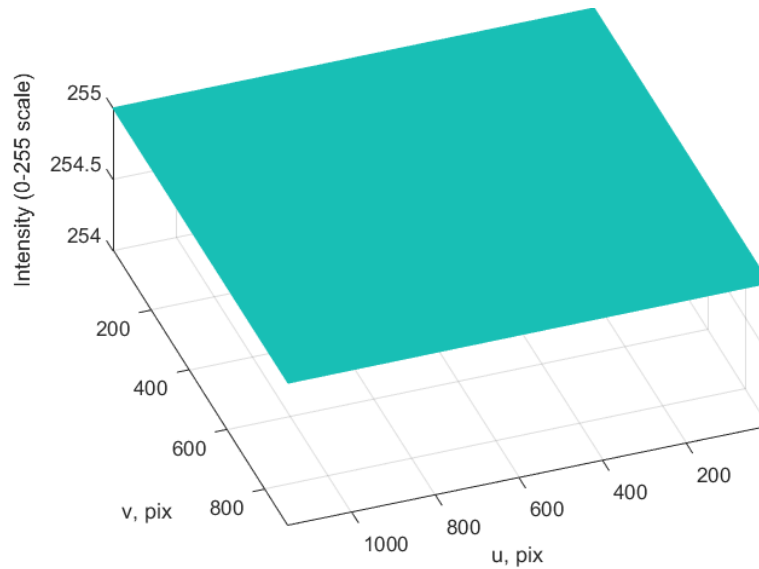


Figure 28. Intensity Plot of Background-Removed Image

The YCbCr masking filter used in earlier experiments with the video in color was used to remove the background noise from the IR image.

The output of the masked image is shown in Figure 29; observe that the timestamp at the top left corner and the skyline at the top edge of the image have been removed. However, the bright spots as well as a small part of the shadow near the bow of the ship were not successfully removed. The mast at the bow of the ship was inadvertently removed as a result of the filtering because the luminance of the mast was much darker than the other parts of the ship. The mast on the bow was not as tall as the mast on the pilothouse; therefore, it did not skew the overall height of the ship despite being removed.

These bright spots near the ship pose a problem when performing bottom-hat filtering to enhance the contrast of the image as the pixels adjoin to the ship. As a result, subsequent edge detection results in a ship that is larger than the actual size. The result after adjusting the contrast of the image is shown in Figure 30.

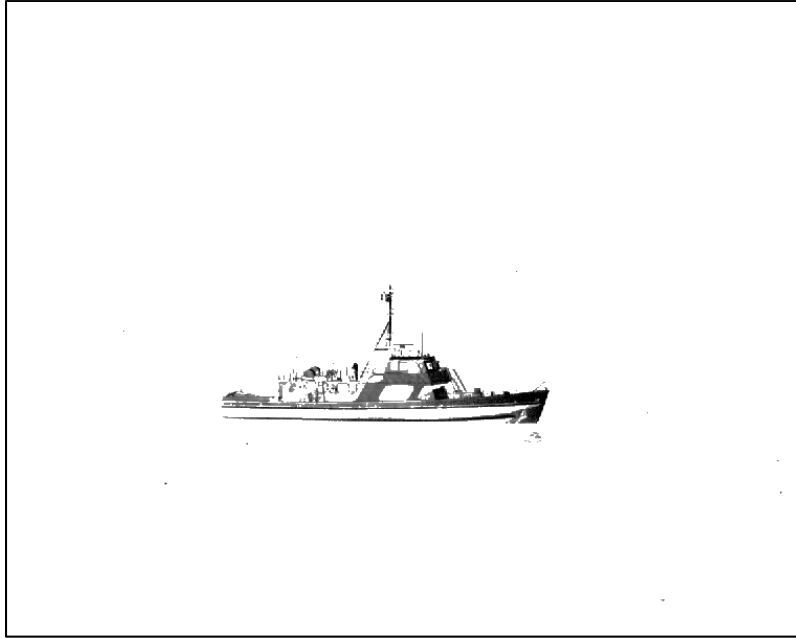


Figure 29. Output from YCbCr Masking

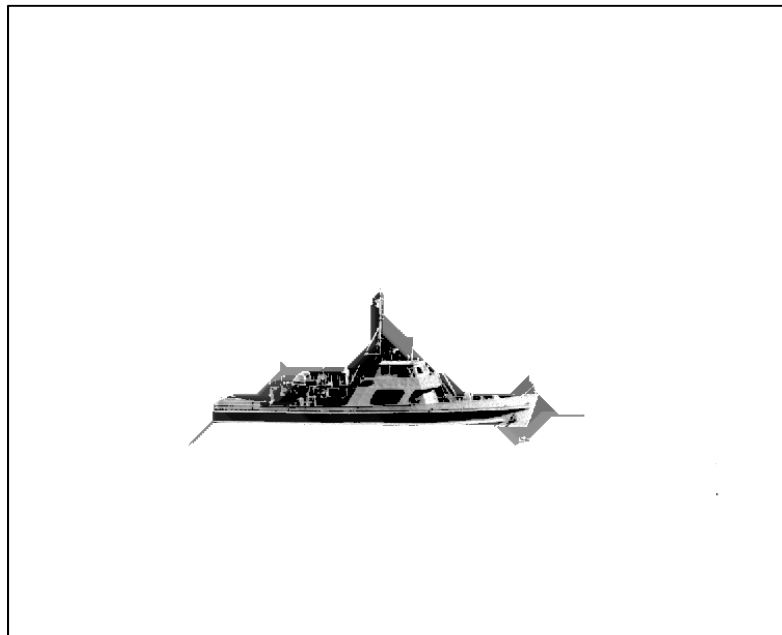


Figure 30. Image after Applying Contrast Adjustment

The image processing routine was then applied to the contrast-enhanced image to perform the steps described in Chapter IV, Section A. The snapshots of each step in the algorithm routine are shown in Figure 31.

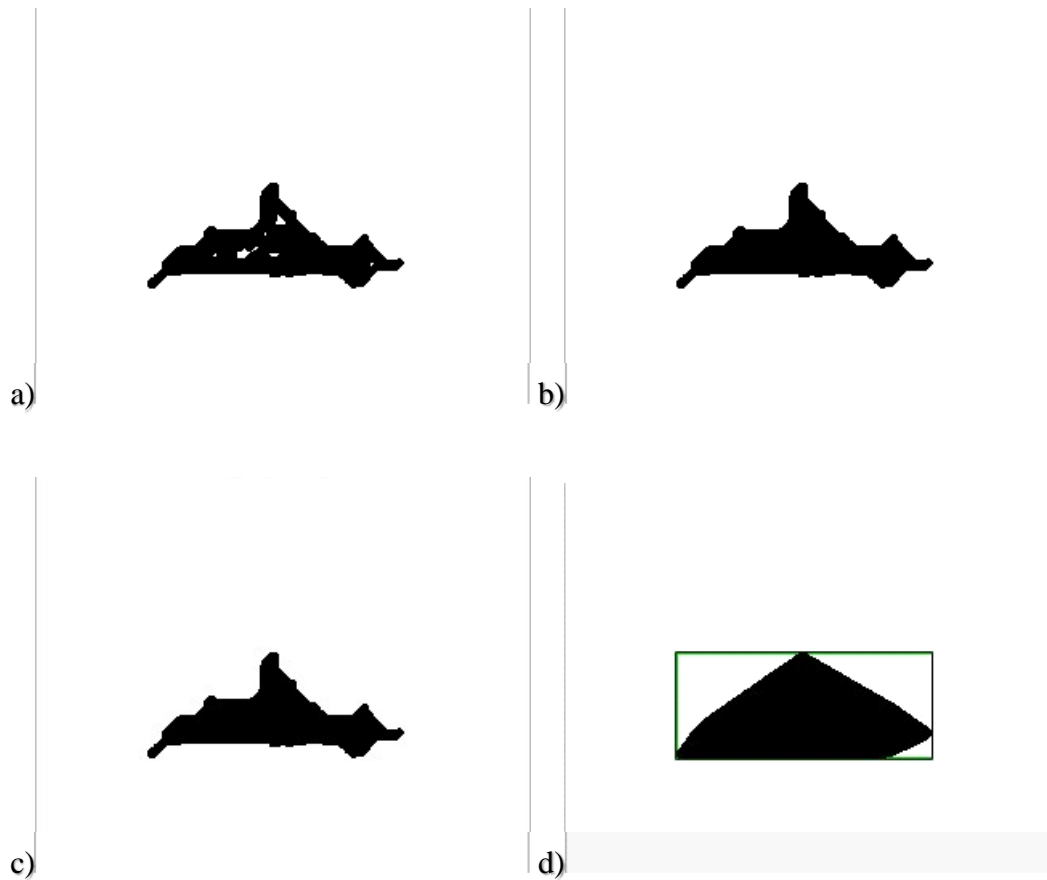


Figure 31. IR Image Processing: Dilation (a), Filling the Holes (b), Clearing Borders (c), Creating a Convex Hull (d)

The bounding box ratio plot is shown in Figure 32. From the plot, it can be seen that the first 14 frames have approximately the same values. From frame 15 onward, the ratio starts to increase although the ship is not turning in this video. As indicated by the plot in Figure 32, the maximum bounding box ratio value corresponds with frame 19 of the video, shown in Figure 33. From the image, it can be observed that the bounding box extends beyond the bow of the ship. This is due to a bird, represented by a group of bright pixels, which the algorithm has mistaken as part of the ship during the image processing. The algorithm was tested with different filtering values to remove this noise (pixels); however, it also removed pixels that constituted part of the ship, thus skewing the bounding box ratio as well.

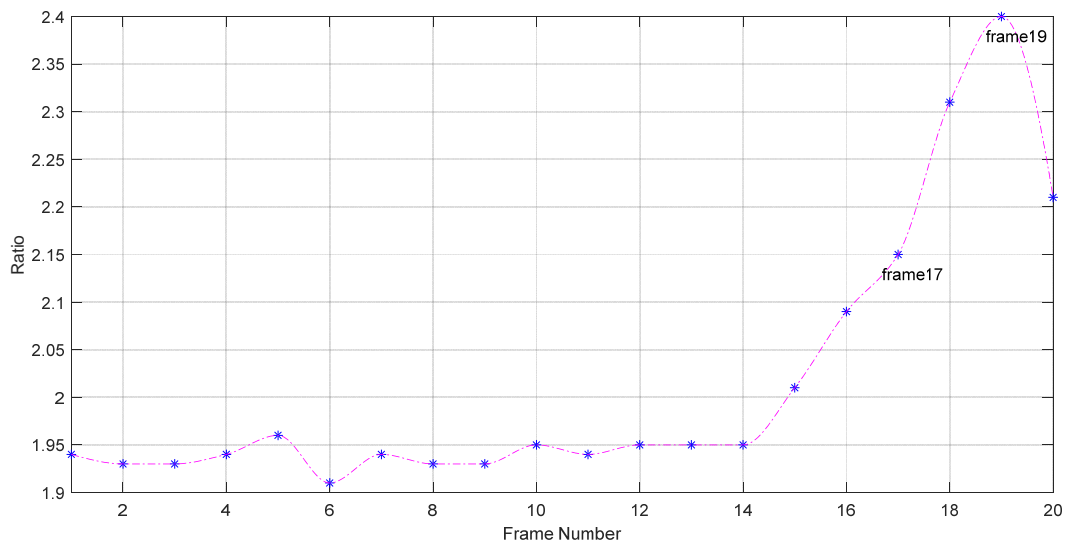


Figure 32. Bounding Box Ratio Plot for IR_Video1



Figure 33. Frame 19 of IR_Video1

(2) Processing IR_Video2

A different video was used to test the same algorithm developed in Chapter IV, Section C. A set of frames extracted from the video is shown in Figure 34. In this video, the shoreline as well as the horizon line can be seen at the top of the image.

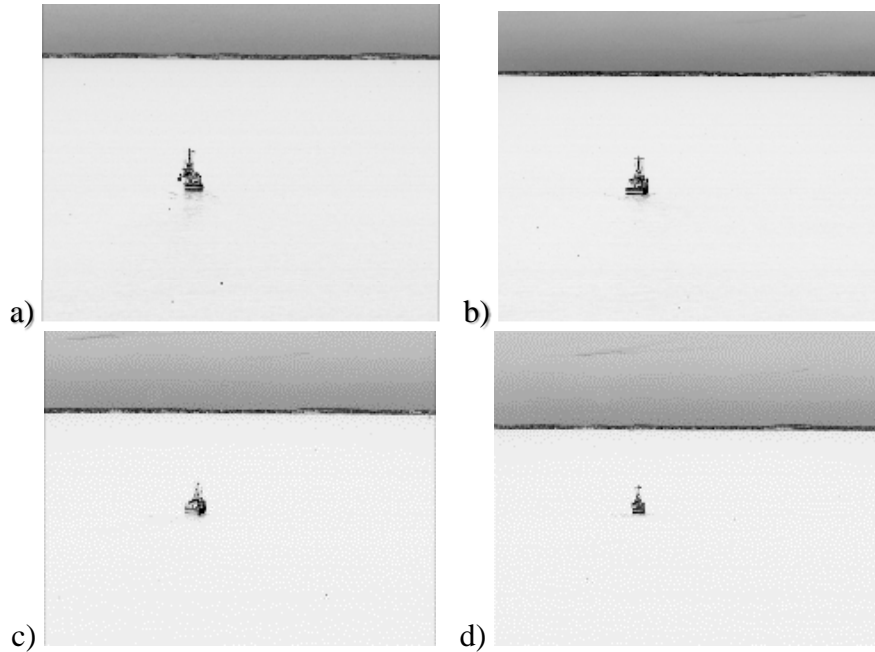


Figure 34. Images 1 (a), 50 (b), 150 (c), and 250 (d) from IR_Video2

The images were passed through the YCbCr filter to remove the background noise. The output from the filter is shown in Figure 35. It can be observed that the YCbCr masking did not completely remove the shoreline nor the horizon line. The contrast of the image was enhanced before the image is passed through the image processing routine. The resultant image is shown in Figure 36.

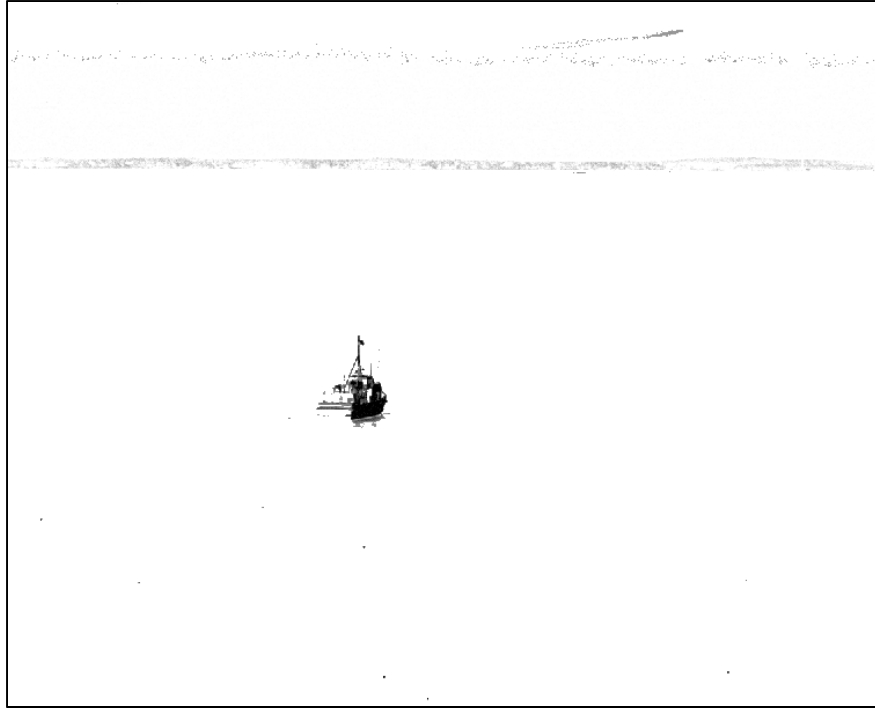


Figure 35. Output from YCbCr Masking of IR_Video2 Image

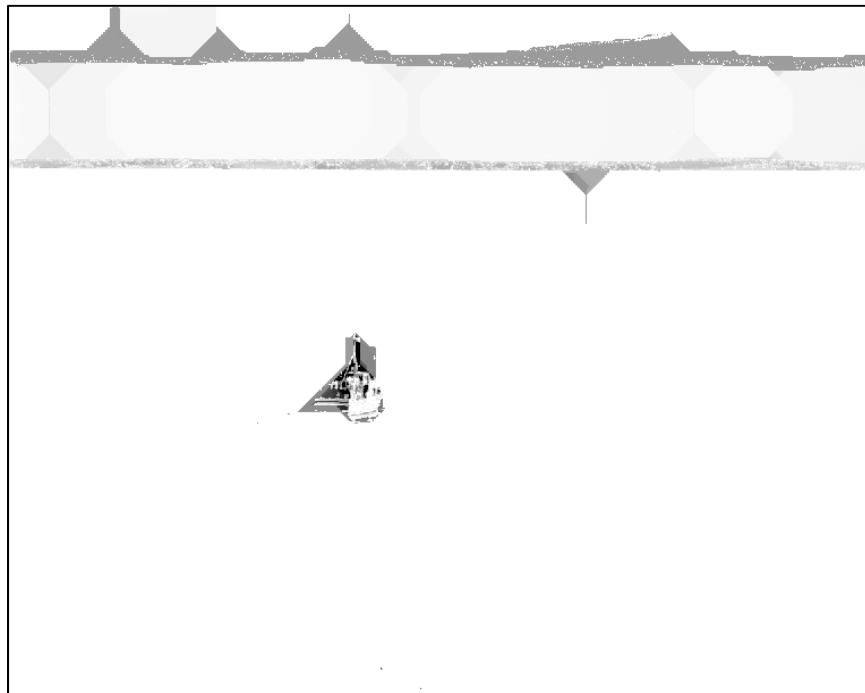


Figure 36. IR_Video2 Image after Contrast Adjustment

The image passed through the image processing routine after the contrast was enhanced. The output from each step of the image processing routine is shown in Figure 37. Notably, the shoreline and horizon line are removed after the clear border operation.

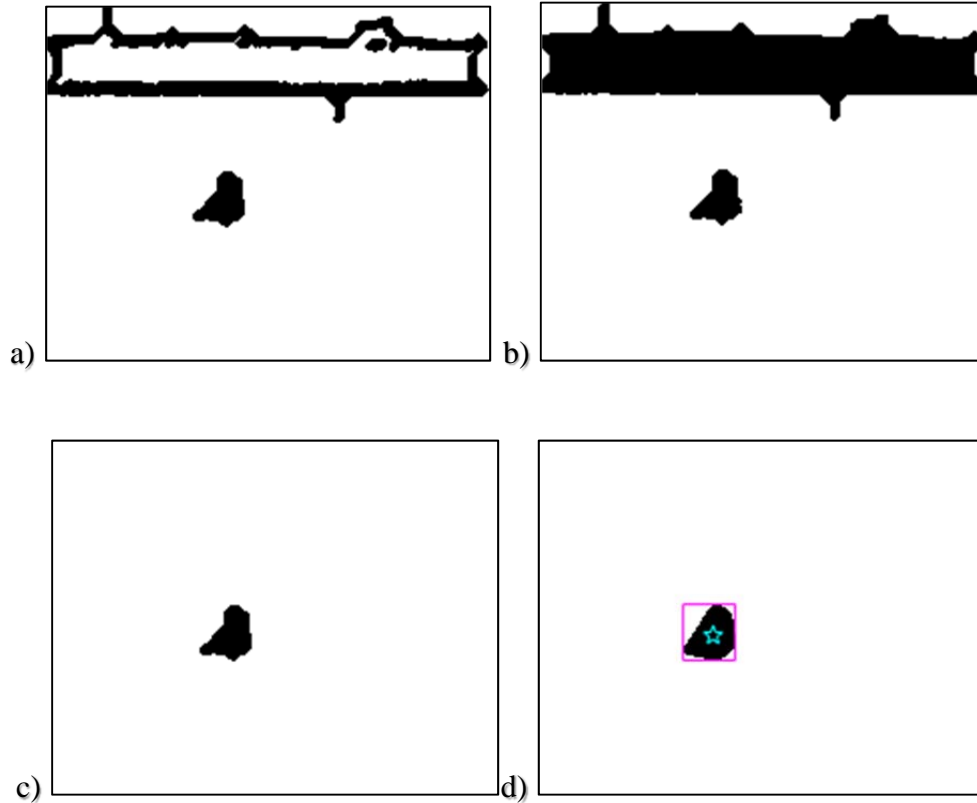


Figure 37. Output from Each Step of Image Processing Routine: Dilation (a), Filling the Holes (b), Clearing Borders (c), Creating a Convex Hull (d)

The bounding box ratio plot is shown in Figure 38. As shown on the plot, an outlier point corresponds with frame number 107. The high bounding box ratio from that particular frame was caused by the processing routine failing to remove the shoreline and horizon line after the clear border operation, as shown in Figure 39. Hence, it caused the processing algorithm to incorrectly identify the shoreline as the ship.

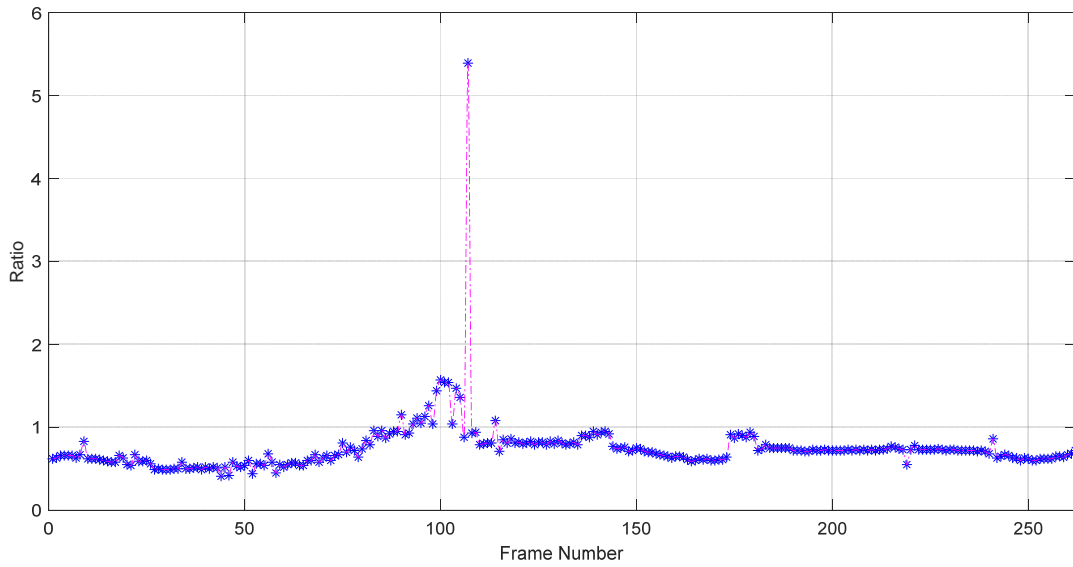


Figure 38. Bounding Box Ratio Plot for IR_Video2

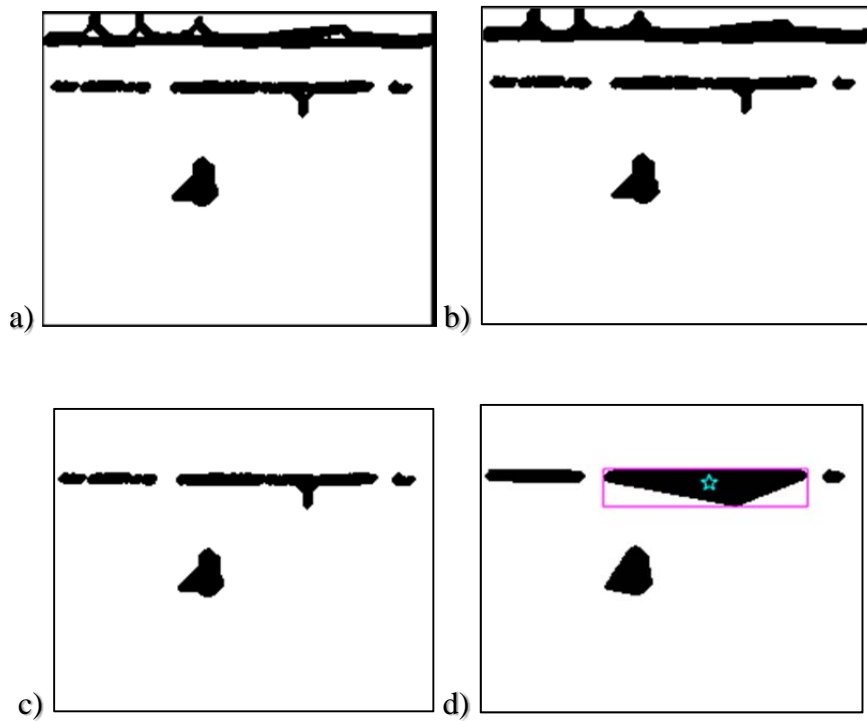


Figure 39. Output from Each Step of Image Processing Routine for Frame Number 107: Dilation (a), Filling the Holes (b), Clearing Border (c), Creating a Convex Hull (d)

(3) Processing IR_Video1 without contrast enhancement

The images extracted from IR_Video1 were processed using the same algorithm without contrast enhancement. As shown by the bounding box ratio plot in Figure 40, there is more variability in between the frames as compared to the plot in Figure 32. The maximum value corresponds with frame 2 of the video while the minimum values correspond to frames 4, 13, and 17 of the video.

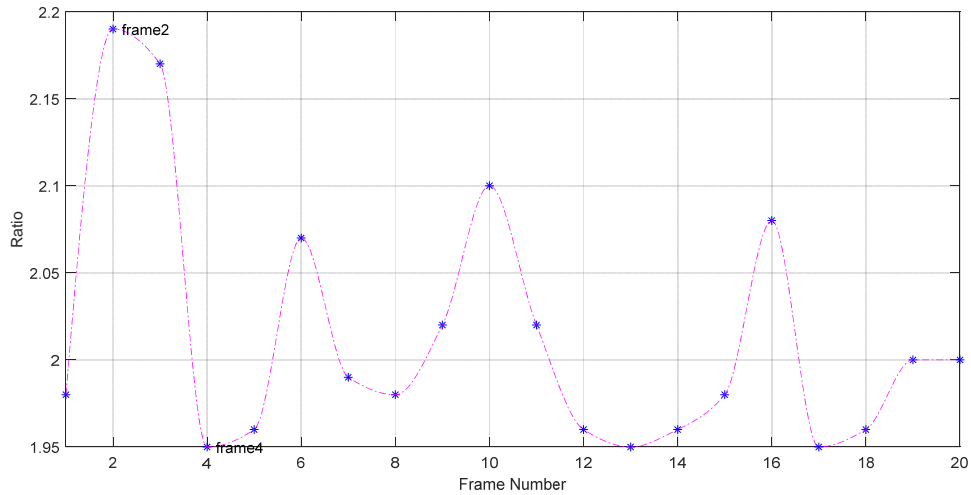


Figure 40. Bounding Box Ratio Plot for IR_Video1 without Contrast Enhancement

As illustrated in Figure 41, the bounding box has the closest fit to the ship. In Figure 42, on the other hand, the bounding box includes parts of the shadow of the ship in water. Notably, Figure 43 does not depict bright spots in the image as Figure 33 does.



Figure 41. Frame 2 from IR_Video1

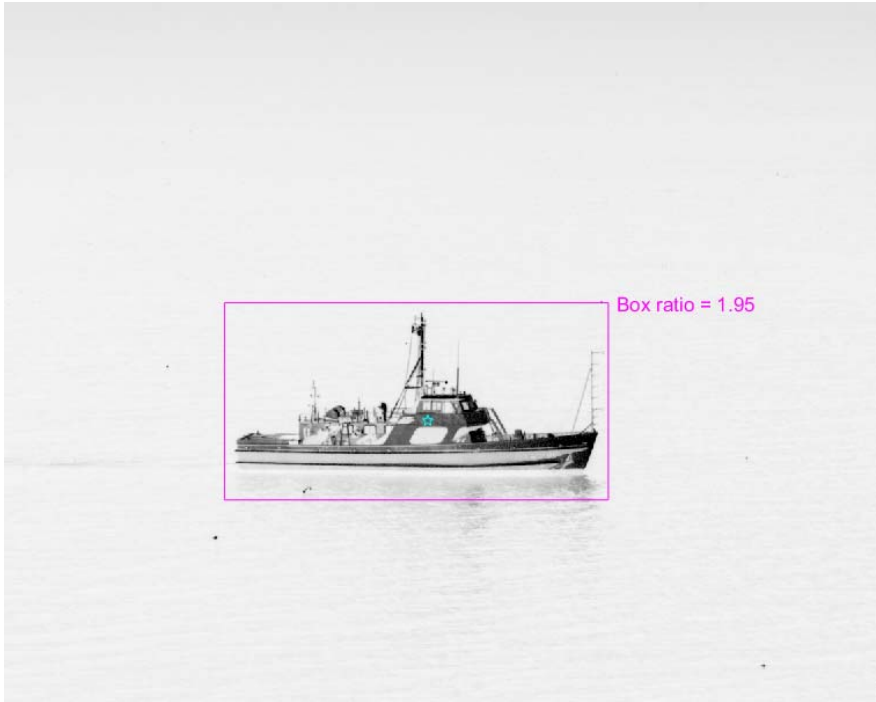


Figure 42. Frame 4 from IR_Video1

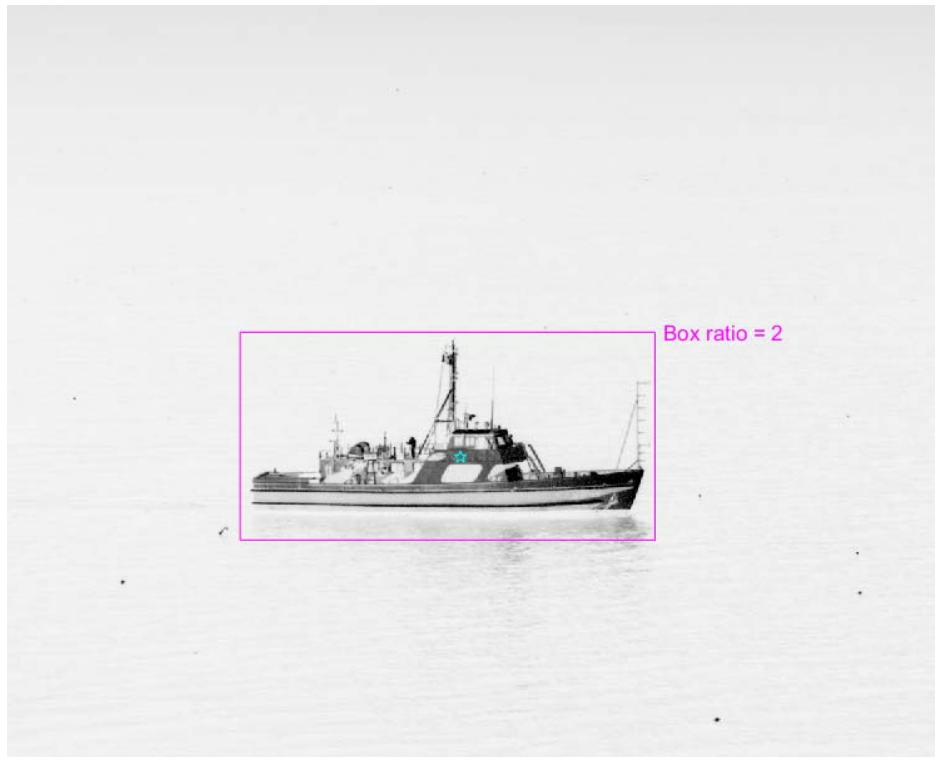


Figure 43. Frame 19 of IR_Video1 without Contrast Enhancement

(4) Processing IR_Video2 without Contrast Enhancement

The same algorithm without contrast enhancement was tested with images extracted from IR_Video2. The bounding box ratio plot is shown in Figure 44. Obviously, one of the frames has a much higher value than the rest. The reason for this high value is that the algorithm treated the wake behind the ship as part of the vessel, as shown in Figure 45.

As illustrated in Figure 44, the value of the bounding box increases between frames 30 and 90, which correspond with the ship turning starboard as it sails away from the camera, as shown in Figure 46.

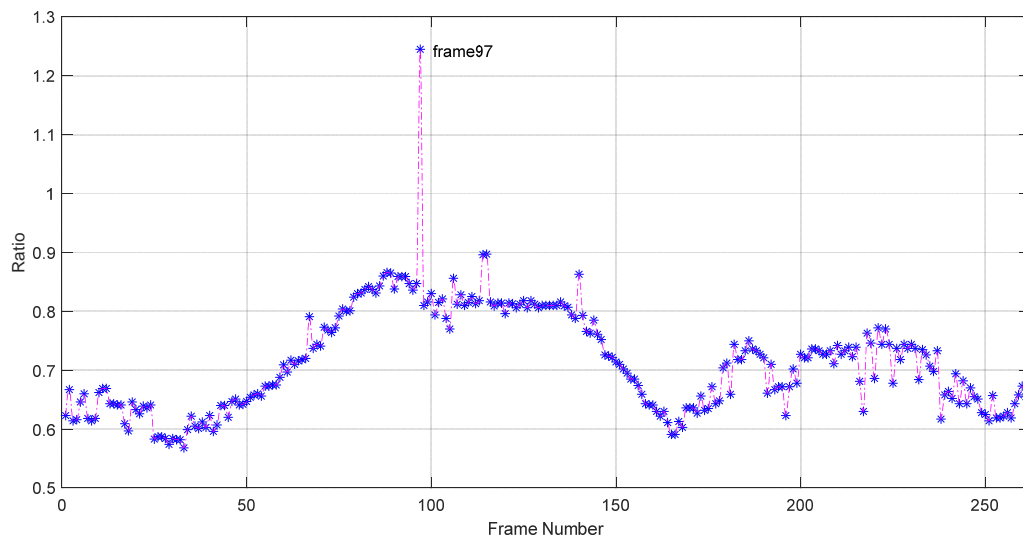


Figure 44. IR_Video2 Bounding Box Ratio Plot without Contrast Enhancement

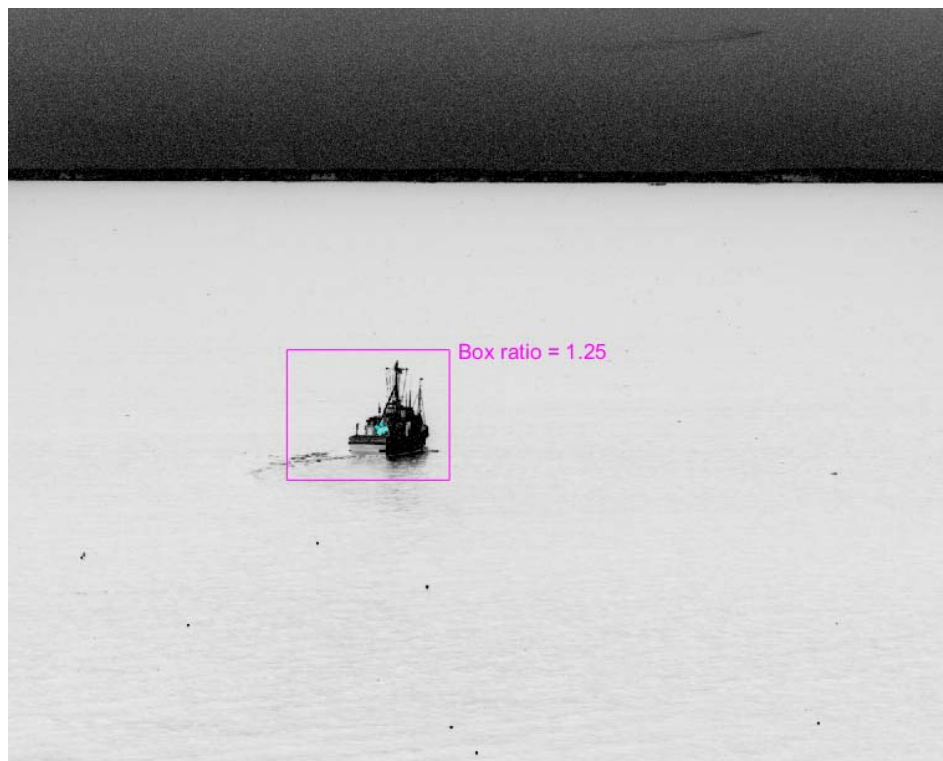


Figure 45. IR_Video2 Frame 97 Bounding Box Image

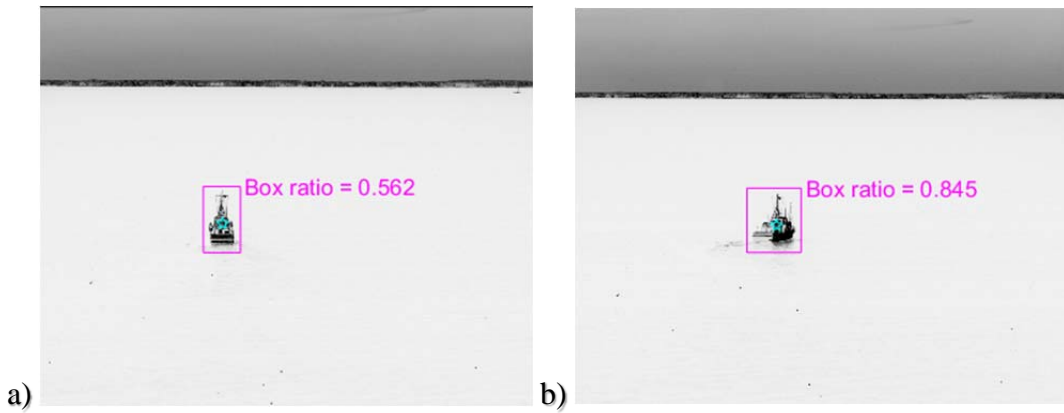


Figure 46. IR_Video2 Frames 33 (a) and 94 (b) with Bounding Box Superimposed

Apart from deriving information from the bounding box ratio, information can also be derived from the bounding box height data. The bounding box height data plotted against each video frame is shown in Figure 47. The plot shows the bounding box height decreasing as the frame number increases, which implies the ship is sailing away from the camera. Therefore, the bounding box height data can be used to infer whether the ship is sailing away from or toward the camera. In Figure 47, frame 114 has a much lower value than the preceding frame. A comparison of the two frames with their bounding boxes is shown in Figure 48. From Figure 48, it is obvious that frame 114 is much darker than frame 113; thus, the algorithm did not detect the mast of the ship, resulting in a much lower value for the bounding box height.

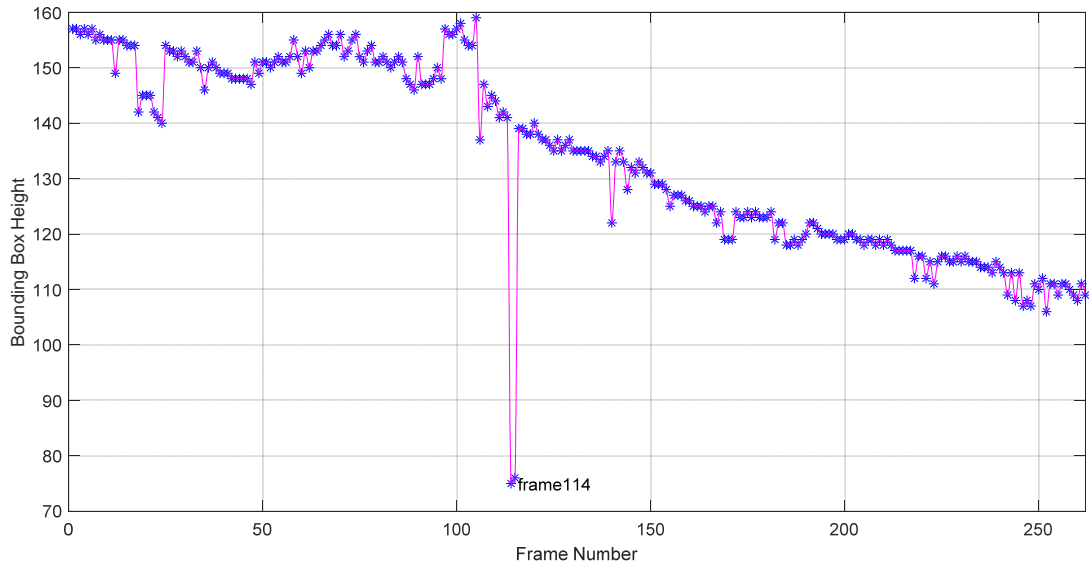


Figure 47. Plot of Bounding Box Height against Video Frames

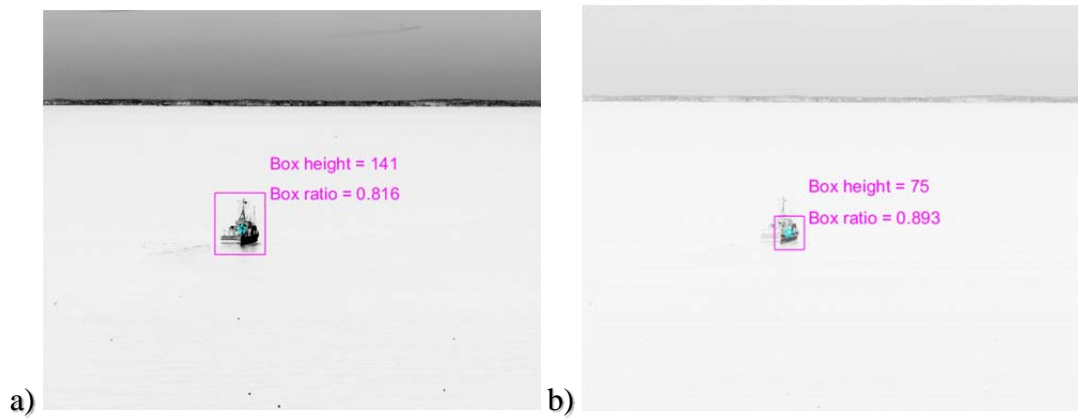


Figure 48. Comparison of Bounding Box Height over Consecutive Frames 113 (a) and 114 (b)

THIS PAGE INTENTIONALLY LEFT BLANK

VI. CONCLUSION AND RECOMMENDATIONS

A. CONCLUSION

This research demonstrated the feasibility of using a computer vision-based technique to derive relevant information to provide a situational awareness (SA) capability for the unmanned surface vehicle (USV).

The needs for developing an algorithm to provide this capability were illustrated using a systems engineering approach. A combination of functional decomposition and functional flow was used to define the algorithm's necessary functions to provide the SA capability.

An image processing algorithm was developed in MATLAB to process video images to derive information that is relevant to the SA capability of the USV. The algorithm attempts to draw a bounding box around a ship detected in the video and subsequently use the characteristics of the bounding box to infer information about the ship's orientation and motion. Different techniques were tested during the development of the algorithm to remove the background noise in the images. It was found that images from color and infrared (IR) videos require different methods to filter out the background noise. Results after filtering show that IR images have less background noise than do color video images.

One of the challenges encountered during the development process of the algorithm was the effect of shadows in the images. The shadows could not be filtered out easily due to their visual similarity to the ship. Because the algorithm falsely detected shadows as part of the ship, the bounding box measurements were skewed. The effect of the shadows appears to be more pronounced when the ship is near the camera. From the experiments performed with the IR video images, the algorithm without contrast enhancement yielded better results.

In conclusion, the bounding box measurements can be used for inferring a ship's orientation and for determining whether it is sailing away from or toward the camera.

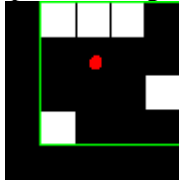
Images from IR video sources were also found more suitable for the algorithm developed in this research as there was less background noise in the image.

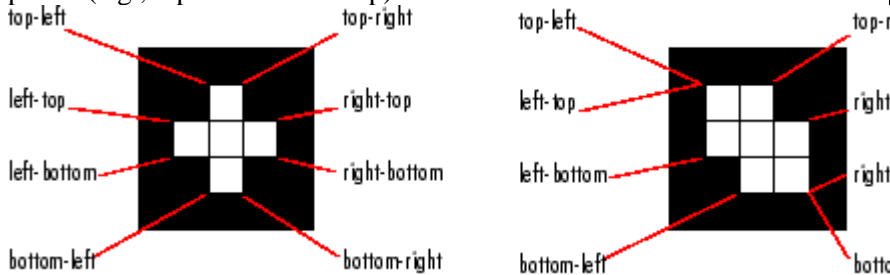
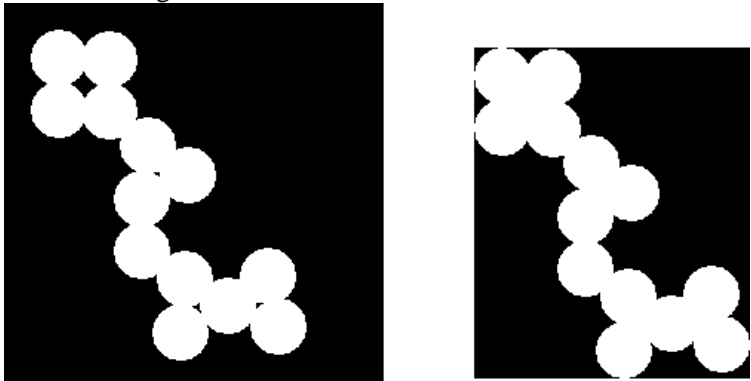
B. RECOMMENDATIONS


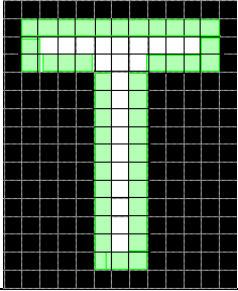
To improve the robustness of the algorithm, more work is required to remove the effects of shadows in the images, so the bounding box measurements can be more accurate. Another area of research ought to involve fusing information derived from the other sensors to provide the USV an SA capability.

APPENDIX. SHAPE PROPERTIES

Shape Measurement Properties in MATLAB. Source: Mathworks (2017d).

Property Name	Description
'Area'	Returns a scalar that specifies the actual number of pixels in the region. (This value might differ slightly from the value returned by <code>bwarea</code> , which weights different patterns of pixels differently.)
'BoundingBox'	Returns the smallest rectangle containing the region, specified as a 1-by-Q*2 vector, where Q is the number of image dimensions, for example, <code>[ul_corner width]</code> . <code>ul_corner</code> specifies the upper-left corner of the bounding box in the form <code>[x y z ...]</code> . <code>width</code> specifies the width of the bounding box along each dimension in the form <code>[x_width y_width ...]</code> . <code>Regionprops</code> uses <code>ndims</code> to get the dimensions of label matrix or binary image, <code>ndims(L)</code> , and <code>numel</code> to get the dimensions of connected components, <code>numel(CC.ImageSize)</code> .
'Centroid'	Returns a 1-by-Q vector that specifies the center of mass of the region. The first element of <code>Centroid</code> is the horizontal coordinate (or x-coordinate) of the center of mass, and the second element is the vertical coordinate (or y-coordinate). All other elements of <code>Centroid</code> are in order of dimension. This figure illustrates the centroid and bounding box for a discontinuous region. The region consists of the white pixels; the green box is the bounding box, and the red dot is the centroid. <div style="text-align: right; margin-top: 10px;">  </div>
'ConvexArea'	Returns a scalar that specifies the number of pixels in 'ConvexImage'.
'ConvexHull'	Returns a p-by-2 matrix that specifies the smallest convex polygon that can contain the region. Each row of the matrix contains the x- and y-coordinates of one vertex of the polygon.
'ConvexImage'	Returns a binary image (logical) that specifies the convex hull, with all pixels within the hull filled in (set to on). The image is the size of the bounding box of the region. (For pixels that the boundary of the hull passes through, <code>regionprops</code> uses the same logic as <code>roipoly</code> to determine whether the pixel is inside or outside the hull.)
'Eccentricity'	Returns a scalar that specifies the eccentricity of the ellipse that has the same second-moments as the region. The eccentricity is the ratio of the distance between the foci of the ellipse and its major axis length. The value is between 0 and 1. (0 and 1 are degenerate cases. An ellipse whose eccentricity is 0 is actually a circle, while an ellipse whose eccentricity is 1 is a line segment.)
'EquivDiameter'	Returns a scalar that specifies the diameter of a circle with the same area as the region. Computed as $\sqrt{4 * \text{Area} / \pi}$.
'EulerNumber'	Returns a scalar that specifies the number of objects in the region minus the number of holes in those objects. This property is supported only for

	2-D label matrices. Regionprops uses 8-connectivity to compute the Euler number measurement. To learn more about connectivity, see Pixel Connectivity.
'Extent'	Returns a scalar that specifies the ratio of pixels in the region to pixels in the total bounding box. Computed as the Area divided by the area of the bounding box.
'Extrema'	<p>Returns an 8-by-2 matrix that specifies the extrema points in the region. Each row of the matrix contains the x- and y-coordinates of one of the points. The format of the vector is [top-left top-right right-top right-bottom bottom-right bottom-left left-bottom left-top]. This figure illustrates the extrema of two different regions. In the region on the left, each extrema point is distinct. In the region on the right, certain extrema points (e.g., top-left and left-top) are identical.</p> 
'FilledArea'	Returns a scalar that specifies the number of on pixels in FilledImage.
'FilledImage'	<p>Returns a binary image (logical) of the same size as the bounding box of the region. The on pixels correspond to the region, with all holes filled in, as shown in this figure.</p>  <p style="text-align: center;">Original Image, Containing a Single Region Image Returned</p>
'Image'	Returns a binary image (logical) of the same size as the bounding box of the region. The on pixels correspond to the region, and all other pixels are off.
'MajorAxisLength'	Returns a scalar that specifies the length (in pixels) of the major axis of the ellipse that has the same normalized second central moments as the region.
'MinorAxisLength'	Returns a scalar that specifies the length (in pixels) of the minor axis of the ellipse that has the same normalized second central moments as the region.
'Orientation'	Returns a scalar that specifies the angle between the x-axis and the major axis of the ellipse that has the same second-moments as the region. The value is in degrees, ranging from -90 to 90 degrees. This figure illustrates

	<p>the axes and orientation of the ellipse. The left side of the figure shows an image region and its corresponding ellipse. The right side shows the same ellipse with the solid blue lines representing the axes, the red dots are the foci, and the orientation is the angle between the horizontal dotted line and the major axis.</p> 
'Perimeter'	<p>Returns a scalar that specifies the distance around the boundary of the region. Regionprops computes the perimeter by calculating the distance between each adjoining pair of pixels around the border of the region. If the image contains discontinuous regions, regionprops returns unexpected results. This figure illustrates the pixels included in the perimeter calculation for this object.</p> 
'PixelIdxList'	<p>Returns a p-element vector that contains the linear indices of the pixels in the region.</p>
'PixelList'	<p>Returns a p-by-Q matrix that specifies the locations of pixels in the region. Each row of the matrix has the form [x y z ...] and specifies the coordinates of one pixel in the region.</p>
'Solidity'	<p>Returns a scalar specifying the proportion of the pixels in the convex hull that are also in the region. Computed as Area/ConvexArea.</p>
'SubarrayIdx'	<p>Returns a cell array that contains indices such that L(idx{:}) extracts the elements of L inside the object bounding box.</p>

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

- Department of Defense. 2011. *Unmanned Systems Integrated Roadmap FY2011–2036*. Reference No. 11-S-3613. Washington, DC: Department of Defense.
- Department of the Navy. 2007. *The Navy Unmanned Surface Vehicle (USV) Master Plan*. Washington, DC: Department of the Navy.
<http://www.navy.mil/navydata/technology/usvmppr.pdf>.
- Endsley, Mica R., and Debra G. Jones. 2011. *Designing for Situation Awareness: An Approach to User-Centered Design*. 2nd ed. Boca Raton, FL: CRC Press, 2011. doi:10.1201/9780203485088.
- Kor, Kian Beng. 2017. “SAF Confident of Coping with Tighter Manpower Resources, Thanks to Increased Automation and Motorisation: Ng Eng Hen.” *Straits Times*, June 30. <http://www.straitstimes.com/singapore/saf-confident-of-coping-with-tighter-manpower-resources-thanks-to-increased-automation-and>.
- Mathworks. 2017a. “Edge Detection Methods for Finding Object Boundaries in Images.” Accessed August 22. <https://www.mathworks.com/discovery/edge-detection.html>.
- . 2017b. “Imfill.” Accessed August 22. <https://www.mathworks.com/help/images/ref/imfill.html>.
- . 2017c. “Morphological Dilation and Erosion.” Accessed August 22. <https://www.mathworks.com/help/images/morphological-dilation-and-erosion.html?requestedDomain=www.mathworks.com>.
- . 2017d. “Regionprops.” Accessed August 22. https://www.mathworks.com/help/images/ref/regionprops.html?s_tid=doc_ta.
- Republic of Singapore Navy. 2017. “Protector: Unmanned Surface Vessel.” Accessed August 22. <https://www.mindef.gov.sg/navy/careers/our-assets/protector-unmanned-surface-vessel.html>.
- Roberts, G. N., and R. Sutton. 2006. *Advances in Unmanned Marine Vehicles*. London: Institution of Electrical Engineers. doi:10.1049/PBCE069E.
- Sobel, I. 1990. “An Isotropic 3 by 3 Image Gradient Operator.” *Machine Vision for Three-Demensional Sciences* 1(1): 23–34. <http://ci.nii.ac.jp/naid/10018992790/>.
- Wong, Kelvin. 2017. “IMDEX 2017: ST Electronics Venus 16 USV Development Enters the Final Lap.” *Jane’s HIS Markit*, May 17. <http://www.janes.com/article/70489/imdex-2017-st-electronics-venus-16-usv-development-enters-the-final-lap>.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
Ft. Belvoir, Virginia
2. Dudley Knox Library
Naval Postgraduate School
Monterey, California