Autonomous Navigation, Perception and Probabilistic Fire Location for an Intelligent Firefighting Robot

Jong-Hwan Kim

Dissertation submitted to the faculty of the Virginia Polytechnic Institute and State University in partial fulfillment of the requirements for the degree of

Doctorate of Philosophy
In
Mechanical Engineering

Brian Y. Lattimer
Alfred L. Wicks
Craig Woolsey
Dennis Hong
Kevin Kochersberger

September 17, 2014
Blacksburg, VA

Keywords: firefighting robot, autonomous obstacle avoidance, fire locating, fire smoke and reflection classification, probabilistic classification

Autonomous Navigation, Perception and Probabilistic Fire Location for an Intelligent Firefighting Robot

Jong-Hwan Kim

ABSTRACT

Firefighting robots are actively being researched to reduce firefighter injuries and deaths as well as increase their effectiveness on performing tasks. There has been difficulty in developing firefighting robots that autonomously locate a fire inside of a structure that is not in the direct robot field of view. The commonly used sensors for robots cannot properly function in fire smoke-filled environments where high temperature and zero visibility are present. Also, the existing obstacle avoidance methods have limitations calculating safe trajectories and solving local minimum problem while avoiding obstacles in real time under cluttered and dynamic environments. In addition, research for characterizing fire environments to provide firefighting robots with proper headings that lead the robots to ultimately find the fire is incomplete.

For use on intelligent firefighting robots, this research developed a real-time local obstacle avoidance method, local dynamic goal-based fire location, appropriate feature selection for fire environment assessment, and probabilistic classification of fire, smoke and their thermal reflections. The real-time local obstacle avoidance method called the weighted vector method is developed to perceive the local environment through vectors, identify suitable obstacle avoidance modes by applying a decision tree, use weighting functions to select necessary vectors and geometrically compute a safe heading. This method also solves local obstacle avoidance problems by integrating global and local goals to reach the final goal. To locate a fire outside of the robot field of view, a local dynamic goal-based 'Seek-and-Find' fire algorithm was developed by fusing long wave infrared camera images, ultraviolet radiation sensor and Lidar. The weighted vector method was applied to avoid complex static and unexpected dynamic obstacles while moving toward the fire. This algorithm was successfully validated for a firefighting robot to autonomously navigate to find a fire outside the field of view.

An improved 'Seek-and-Find' fire algorithm was developed using Bayesian classifiers to identify fire features using thermal images. This algorithm was able to discriminate fire and

smoke from thermal reflections and other hot objects, allowing the prediction of a more robust heading for the robot. To develop this algorithm, appropriate motion and texture features that can accurately identify fire and smoke from their reflections were analyzed and selected by using multi-objective genetic algorithm optimization. As a result, mean and variance of intensity, entropy and inverse difference moment in the first and second order statistical texture features were determined to probabilistically classify fire, smoke, their thermal reflections and other hot objects simultaneously. This classification performance was measured to be 93.2% accuracy based on validation using the test dataset not included in the original training dataset. In addition, the precision, recall, F-measure, and G-measure were 93.5 – 99.9% for classifying fire and smoke using the test dataset.

Dedication

For Rosana K. Lee and Leah R. Kim, you have brought me more happiness than I ever thought possible. Your patience as I pursued my dreams puts me forever in your debt.

For my parents, brothers and sisters, you have supported me throughout my entire academic career, from kindergarten before I had any idea of what I could become to this moment which I have dreamt to achieve for so long. Thank you.

Acknowledgements

I would like to thank the following people:

- Dr. Lattimer for your continual guidance, patience, mentoring, and great support throughout the entire PhD life. It has been a long uphill trek which would have never been completed had it not been for your great help.
- Dr. Wicks, Dr. Hong, Dr. Woolsey, and Dr. Kochersberger for support, guidance, and patience along the way to accomplishing this goal. It is safe to say without your guidance has been invaluable in completion of this work.
- Joseph Starr and Josh McNeil for your friendship, advice, and the many discussions, both research related and non-research related, to help pass the time and turn this experience into a fruitful one in so many ways.
- Jackie Woodyard, Cathy Hill, Ben Poe, and Jamie Archual for working on all of the behind the scenes pieces to make sure the research continues without a hitch.

하면 된다!

Table of Contents

List of Figures

List of Tables

1 Introduction

1.1 Motivation

Firefighting is a strenuous and dangerous job that puts firefighters at risk. In 2011, 70,090 firefighters in the U.S. alone were injured in the line of duty with 61 deaths [1-4]. Over 60% of the firefighter deaths and over 20% of the firefighting injuries are caused by exposure to fire conditions such as smoke inhalation, burns, overexertion/stress, or being trapped [3-5]. The visibility conditions that develop inside of structure fires as well as wildland fires can range from relatively clear to zero visibility. The zero visibility created by the presence of smoke influences human behavior such as initial response, redirection of movement, and walking speed [6-9] in addition to making firefighting difficult [3, 4]. Although there has been research on auditory and tactile perception for the purpose of evacuation under fire smoke environments [10, 11], these techniques are not sufficient to support firefighting tasks in a timely manner. Effective technology is needed to allow robots and firefighters to accurately image and navigate through zero visibility smoke to rapidly find fires and victims as well as establish escape routes if conditions begin deteriorating. In order to reduce loss of firefighters and increase their effectiveness on the job, firefighting robots are beginning to be considered.

The robots in Table 1 have been developed to assist firefighters or perform firefighting by autonomous navigation and through remote control. Japan developed Rainbow 5 [12], an unmanned robot that has the capability to be used in large fires such as petrochemical complex fires, aircraft crash fires, etc. This robot has an obstacle remover capable of discarding fallen or dangerous objects. The robot LUF60 [13] was designed in Germany to exhaust gas and extinguish fires from sixty meters away. F2GV [13], designed in Malaysia, has a track system with heavy rubber track belts. It also has a maximum speed of 2.36 km/h and curb weight of 910 kg. A water hose mounted on the top of the robot is controlled remotely to suppress fire. Unfortunately, these robots were designed to perform outdoor firefighting tasks and are too large to be utilized indoors.

Table 1. Firefighting robots

Robot	Type	Application	Operation	Perception	Fire Location	Country	Reference
AFFMP	Mobile robot	Indoor	Autonomous	Flame, LDR	Fire Detection in FOV	Malaysia	[14]
Anna Konda	Snake-like shape	Indoor	Autonomous	Visual camera Thermometer	Fire Detection in FOV	Norway	[15]
Archibot	Ground vehicle	Indoor	Remote control	Visual camera	By operator	South Korea	[16]
F2GV	Ground vehicle	Outdoor	Remote control	Visual camera	By operator	Malaysia	[13]
FINE	Mobile robot	Indoor	Autonomous	Infrared Thermometer	Fire Detection in FOV	Undisclosed	[16]
Fire Search	Mobile robot	Indoor	Remote control	Visual camera	By operator	Japan	[12]
Fire Searcher	Mobile Robot	Indoor	Remote control	Near IR camera Visual camera	By operator	South Korea	[17]
Firemote® 4800	Ground vehicle	Outdoor	Remote control	Visual camera	By operator	China	[16]
FRIGO-M	Mobile robot	Indoor	Remote control	Visual camera	-	Japan	[16]
Hoyarobot	Mobile robot	Indoor	Autonomous	Thermometer	By operator	South Korea	[18]
Jet Fighter	Ground vehicle	Outdoor	Remote control	Visual camera	By operator	Japan	[12]
LUF 60	Ground vehicle	Outdoor	Remote control	Visual camera	By operator	Germany	[13]
OLE	Bug-like shape	Outdoor	Autonomous	Thermometer	Fire Detection in FOV	Germany	[16]
QinetiQ	Mobile robot	Indoor	Remote control	Visual camera	By operator	United Kingdom	[19]
Rainbow 5	Ground vehicle	Outdoor	Remote control	Visual camera	By operator	Japan	[12]
SAFFiR	Humanoid	Indoor	Autonomous	Visual camera Thermal IR stereo	Fire location outside FOV	United States	Current Study
SWARM	Mobile robot	Indoor	Autonomous	No perception instrument	Fire Detection in FOV	United Kingdom	[20]

There are several firefighting robots applicable to indoor tasks. In South Korea, Fire Searcher [12], a scout robot is sent to hazardous sites where fire and poisonous gases exist. This robot delivers internal conditions about fire and victims inside of the building. The operator can directly communicate with the trapped victims via wireless communication, allowing for an efficient rescue plan. Tehzeeb [21], from Thailand, is a semi-autonomous robot designed to rescue victims. This robot can localize itself and build a map of its surroundings as it searches. QinetiQ [19], developed in the United Kingdom has vision sensors that allow the operator to monitor the situation and remotely operate the robot. It is also equipped with a single arm to open doors and handle dangerous objects that may be too hot and/or too heavy for the firefighter to lift as well as objects that may be potentially explosive. Firemote 4800 [16] in China has thermal and visual image cameras that allow the operator to monitor the fire situation. Firemote 4800 uses water stored in its body to suppress fire. With its water reservoir and own cooling system, the robot is also able to maintain reasonably low body temperatures during tasks in high temperatures. Archibot [16], currently in usage in South Korea, is another remotely controlled robot. This robot delivers thermal and visual images to the operator. This robot, similarly to Firemote 4800, has a cooling system as well as a fire suppression system in its body.

All the above mentioned robots require an operator to remotely control their movements, which is the method used for the majority of the current firefighting robots. A major advantage of the remote control is that an operator can observe the situation and make decisions from a safe area [17]. Adding human intelligence to robotic functionalities is another great advantage of the remote controlled firefighting robots. However, there are limitations. First, it is best that one operator control one robot at a time. If an operator controls several robots simultaneously, the operator may not execute the necessary actions needed for each robot in a timely fashion [16]. Secondly, the operator must rely solely on the limited information the robot provides during a given situation. Thirdly, limitations of the current wireless communication technology can cause unstable connections that make data collection and communication challenging. Lastly, remote controlled firefighting robots are only work in progress towards ultimately making firefighting robots fully autonomous.

The following are known autonomous firefighting robots. Autonomous Fire Fighting Mobile Platform (AFFMP) [14] from Malaysia is designed to detect flame and extinguish it by

using a fan mounted on top. A set of black lines guides its patrolling movement using a line following technique. Another from Korea is the portable fire evacuation guide robot, Hoyarobot [18], which can be remotely controlled and was built to be thrown into high risk areas to transfer environmental information back to the operators. In addition, Hoyarobot is capable of searching for victims and helping them evacuate by voice communication between firefighters and the victims themselves. Another is FINE [16]; this has the ability to avoid obstacles with two proximity sensors and find fire with two IR thermometers. This robot was originally designed to extinguish small house fires. Specific details of this robot are unknown and concealed as it is currently being developed for commercial use. SWARM [20] from the UK is a set of robots that is currently under research. During dense smoke, these robots are to be positioned near the firefighter and help by providing information about its surroundings. However, this robot is only capable of using a basic level navigation algorithm. Recently, a shipboard autonomous firefighting robot (SAFFiR), which is the first autonomous humanoid firefighting robot and its prototypes are illustrated in Figure 1 and Figure 2, is being developed at Virginia Tech to locate and suppress fires inside ships and structures. Through the SAFFiR program, further advancements of artificial intelligent algorithms / perception systems [22-26] and unmanned fire suppression systems [27] have been developed to enhance autonomous firefighting robots. Despite considerable research and efforts on autonomous firefighting robots, real-time navigation in complex environments with dynamic obstacles and real-time goals has not been fully accomplished. However, locating a fire inside of a structure that is not in the direct field of view (FOV) of the robot has been accomplished.

Figure 1. A prototype of Shipboard Autonomous Firefighting Robot (SAFFiR)

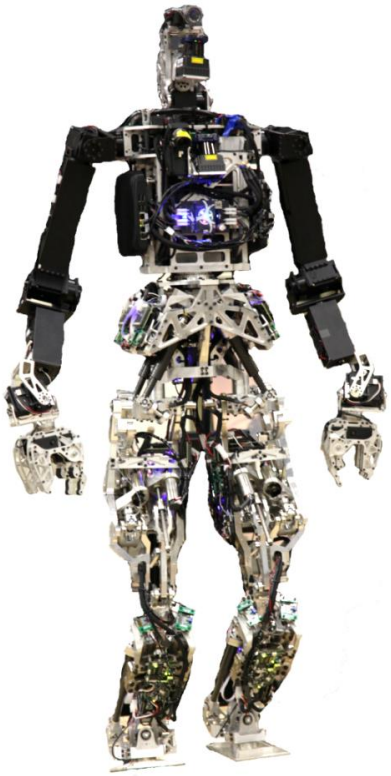


Figure 2. Shipboard Autonomous Firefighting Robot (SAFFiR)

There are challenges in developing intelligent firefighting robots. Fire environments are rapidly changing and deteriorating with complex static and unexpected dynamic obstacles. Fires with dense smoke are high in temperature. Dense smoke produces low visibility, e.g. less than 1.0 m resulting in difficult conditions for navigation. Due to these conditions, sensors commonly used for robotics cannot properly function. Table 2 shows test results of widely used range sensors for robotics tested in light and dense smoke fire environments. One sensor used for range finding in robotics is Lidar (Light Detection and Ranging), which is a rotating laser (0.905 μm wavelength) that uses time of flight of the light to determine the distance to an object. In fire experiments, signals from Lidar (single and multi-echo) and visual cameras operating in the visible to short wavelength infrared (IR) (less than 1 micron) commonly used for robot navigation were determined to be attenuated with a smoke visibility of 4 m or less with no signal at 1 m visibility [25]. This was attributed to emission and absorption of energy from soot particles reducing with an increase in light wavelength resulting in a lower extinction coefficient at higher wavelengths [28-30]. Because visibility varies by the reciprocal of the extinction coefficient [31], the visibility increases for sensors operating at higher wavelengths.

Another commonly used sensor for perception is a time of flight sonar/ultrasonic sensor [32-34]. In fire testing, the sonar sensor was determined to not be able to accurately measure distance through fire smoke because the speed of sound varies as the gas temperature of the smoke changed [25]. The RGB-depth sensor (Kinect^{TM}) also failed to provide 3-D imaging of the scene due to attenuation of the speckle pattern (produced using short wavelength IR light) through smoke as well as the incoming radiation on the sensor from the fire interfering with the return speckle pattern [25]. However, thermal IR cameras that operate in the long wavelength IR range (7.5 to 14 μm) and radar operating at a wavelength of about 11.5 mm performed well in clear and dense fire smoke conditions [25]. As a result, IR cameras and radar were the recommended sensors for perception systems to be used in low visibility fire smoke applications.

Table 2. Range sensor comparison under light and dense smoke [25]

Instrument	**Light-Smoke Fire**	**Dense Smoke Fire** (less than 1.0 m visibility)
Lidar	Functions	Attenuation at 4m visibility; failure at 1m visibility
RGB camera	Functions	Attenuation at 8m visibility; failure at 1m visibility
Kinect™	Sensor flooded by light from fire during whole test	Poor results even with >8m Visibility (Combination of particle blocking and sensor being flooded by light from fire)
Night Vision	Sensor flooded by light from fire during whole test	Failure at about 4m visibility
Long Wave Infrared Camera	Functions	Functions
FMCW Radar	Functions	Functions
Ultrasonic	Attenuation with temperature change	Some attenuation with temperature change

1.2 Objectives

The environments that firefighting robots overcome include complex static and unexpected dynamic obstacles in unknown environments where a pre-map of obstacles and fire sources is not present. In order for firefighting robots to autonomously locate a fire inside of a structure that is not in the direct robot field of view, this study presents autonomous navigation, perception, and real-time probabilistic fire location. This study enables the firefighting robots to analyze local conditions in unknown environments using long wavelength infrared camera images, classify fire, smoke and thermal reflections by characterizing their patterns, compute the safe heading to avoid static and dynamic obstacles while moving toward the fire, and ultimately locate the fire even outside the robot FOV. This study is introduced in the following chapters structured in the form of papers with relevant literature review of each topic. The chapters are organized in the following manner:

- Chapter 2 – A real-time local obstacle avoidance method was developed for use on firefighting robots that navigate through cluttered and dynamic indoor fire-smoke environments by perceiving the local environment through vectors, geometrically calculating a safe heading, and avoiding complex static and unexpected dynamic obstacles. This new approach solves local obstacle avoidance problems by integrating global and local goals to reach the final goal.
- Chapter 3 – A sensor fusion based 'Seek-and Find' fire algorithm was developed and implemented on to an intelligent firefighting mobile robot to autonomously find a fire by fusing long wavelength infrared camera, ultraviolet radiation sensor, and LIDAR. This is the first autonomous fire location algorithm for use indoors reported in the literature. The weighted vector method in Chapter 2 was implemented as well to locate the fire while avoiding static and dynamic obstacles.
- Chapter 4 – Threshold based algorithms to identify fire features used in Chapter 3 resulted in some false headings for the robot. To increase the accuracy of the heading, the best combination of object features (e.g., motion, texture) in thermal images that accurately classifies fire, smoke and their thermal reflections was analyzed and determined by using multi-objectives genetic algorithm optimization. Prior to this work, no research had

investigated the appropriate object features to accurately classify fire, smoke, and thermal reflections from IR images. In addition, the features are applicable for dynamic as well as stationary imaging systems.

- Chapter 5 – A real-time probabilistic classification method was developed to identify fire, smoke, their thermal reflections, and other objects using infrared images. This algorithm was formulated for use on a robot that will autonomously decide a heading to locate fires inside of a structure where the fire is outside the robot field of view. This probabilistic approach for classifying fire objects using IR images or visible images has not been reported in the literature and represents a significant improvement in identifying fire objects, particularly on dynamic platforms.

2 Weighted Vector Method for Real-Time Indoor Obstacle Avoidance in Dynamic Environments

2.1 Abstract

A real-time local obstacle avoidance method is developed in this work for use on firefighting robots that navigate through indoor environments with real-time implementation, safe heading calculation, and avoidance of complex static and unexpected dynamic obstacles. This method perceives the local environment through vectors, classifies suitable obstacle avoidance modes by applying a decision tree, uses weighting functions to select necessary vectors and geometrically computes a safe heading. In addition, this method solves local obstacle avoidance problems by integrating global and local goals to reach the final goal. The method was implemented on a mobile robot platform and demonstrated to successfully navigate through different test environments by sequentially switching the obstacle avoidance modes to overcome obstacles in the surroundings. Furthermore, this method was successfully validated for a firefighting robot to autonomously navigate to find a fire outside the field of view.

2.2 Introduction

Obstacle avoidance is crucial in order for robots to autonomously perform human tasks during disasters such as fires. In fire incidents, obstacle avoidance is made more difficult through low visibility, fast changing environments, suddenly appearing obstacles, and uncertainty of fire location. There has been active research of firefighting robots to reduce the loss of firefighters and increase their effectiveness on the job. Larger robots [12, 13, 16] have been designed to assist in suppressing outdoor fires such as wildland and tanker fires while smaller, more maneuverable robots [16-20] have been developed specifically for indoor fires. However, most of the indoor firefighting robots require an operator to remotely control their movements and actions due to operator safety and use of human intelligence [17]. With remote control, it is

difficult for operators to control several robots simultaneously to ensure timely execution [16] and the operator commonly relies on limited information about the environment to make movement decisions. For many indoor applications, limitations of the current wireless communication technology can cause unstable connections that make data collection and communication challenging. As a result, onboard processing of the environment to make decisions on navigation in areas with many static and sometimes dynamic obstacles is needed to successfully launch robots into fire environments.

Despite considerable research and efforts on autonomous firefighting robots [14, 18, 20, 24, 27], real-time navigation in complex environments with dynamic obstacles and real-time goals has not been fully accomplished. Navigation by computing paths from a pre-existing map and obstacle information [35] is not suitable because the surrounding environment of a structure fire can rapidly change, including obstacles. Methods [36-38] that use center coordinates of obstacles for path-planning calculation have limited practical applicability in unknown environments [39]. To avoid unexpected dynamic obstacles, a firefighting robot must be equipped with a real time algorithm capable of quickly recognizing the unexpected dynamic obstacles and avoiding them. The navigation algorithm must also keep track of the final goal to ensure that the robot is not only avoiding obstacles but also moving toward the goal. As a result, an obstacle avoidance method is needed that can operate in real-time, avoid complex static obstacles, avoid dynamic obstacles, and predict a safe heading while leading toward a goal.

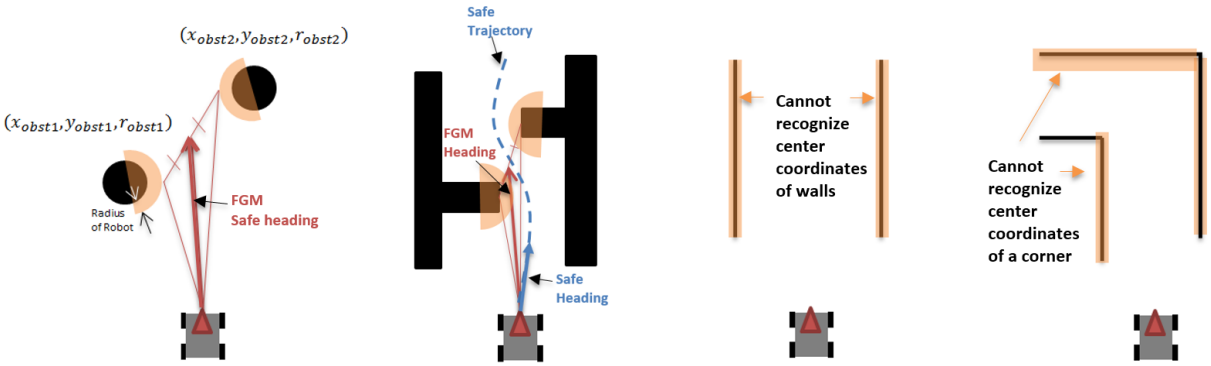
Local obstacle avoidance algorithms are quite suitable for this application. Unlike static path planning that finds geometric paths to the goal using pre-existing information of a map and goal, the local obstacle avoidance calculates paths that avoid obstacles along the trajectory of the robot by frequently perceiving the local environment. The static path planning is useful for optimized path in static environment [35, 40, 41], but is limited in dynamic environments due to the fact that environment information cannot be updated fast enough to alter the planned path [42, 43]. However, since obstacle avoidance algorithms regularly read and respond based on local information, this type of algorithm is suitable in dynamic environments such as fire environments which may rapidly change.

Algorithms have been proposed for local obstacle avoidance. Bug 1 [44] and 2 [45] algorithms simply avoid obstacles by making the robot adhere close to the obstacles, but its trajectory can be inefficient and unsafe under deteriorating structures in fire environments. The

Bubble Band technique [35] creates a safe bubble (a collision-free local space) by calculating the empty space around the robot, but it cannot be implemented in real-time because this method requires environmental information ahead of time. Potential Field Method [46] and Artificial Potential Field [47] view the goal as attractive potential and read obstacles as repulsive potential to avoid obstacles while moving toward the goal. Curvature Velocity Method [38] and Lane Curvature Method [48] take into consideration physical constraints; rotational and translational velocities and add a lane method for local obstacle avoidance. Vector Field Histogram (VFH) [49] and VFH$^+$ [50] map the local environment surrounding the robot while identifying the openings that are wide enough to allow passage. Although the last six algorithms [46-50] are widely applied for autonomous navigation of mobile robots, their obstacle avoidance performance are limited in cluttered and complex indoor spaces along with a local minima problem that causes the robot to stop on the way to its goal [51].

Vector Distance Function method (VDF) [37] utilizes weighted forces to discretize a free-space force for obstacle avoidance in dynamic environments. This approach efficiently avoids dynamic obstacles by sensing the surrounding area and simply weighting the direction the robot needs to move. Although VDF has a fast response suitable for dynamic obstacle avoidance, it is not able to compute safe trajectories which has the maximum distances from obstacles [36] in complex and cluttered spaces. In addition, oscillation in movement may occur when there is not enough information on the surroundings [52]. The Following Gap Method (FGM) [36] computes a safe trajectory to the goal while avoiding static and dynamic obstacles without a local minima problem by constructing a gap array from observable obstacles and calculating an angle toward the center of the maximum gap. Despite these advantages, FGM has several drawbacks some of which make it unsuitable for indoor environments. FGM requires additional computation for transforming between Cartesian and Polar coordinates. FGM becomes aware of an obstacle with x, y and radius of an obstacle in Cartesian coordinates as shown in Figure 3, but it calculates the angle for the robot to move in Polar coordinates. The FGM has limitations computing safe trajectories when the robot travels in complex geometries (e.g. maze) because it computes the direction to the center point of a gap that may not be a safe route as shown in Figure 3 (b). In addition, FGM needs x, y, and radius of an obstacle in order to recognize it properly. However, when the obstacles are continuous such as walls, hallways and a corner in Figure 3 (c) and (d), FGM cannot make the proper assessment because the walls cannot be

simply represented as an obstacle with an x, y, and radius. Lastly, it has not been tested indoors where a large number of obstacles such as walls, hallways, corners, rooms, and maze like environments are present.



(a) general (b) maze-like (c) hallway (d) corner

Figure 3. Obstacle representation and geometry of FGM (a); limitations of FGM in a maze-like environment, hallway (c) and corner (d).

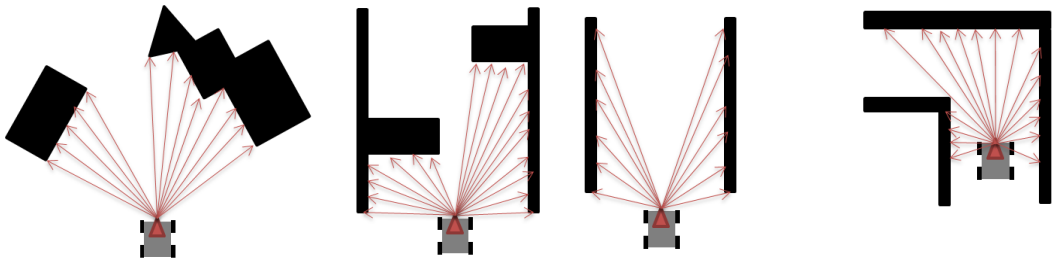
This paper presents a new local obstacle avoidance method called the Weighted Vector Method (WVM). The WVM is a real-time local obstacle avoidance method that is capable of navigating around static and dynamic obstacles while still moving toward a goal. This method recognizes the local environment through vectors, classifies suitable avoidance modes by using a decision tree, uses weighting functions to select necessary vectors, and geometrically calculates the appropriate heading angle. The vector representation and configuration space division overcome the shortcomings of other similar methods such as FGM and VDF, allowing the WVM to be applicable for indoor environments. In addition, WVM is able to perceive objects of any shape and compute safe trajectories in environments that have complex static and unexpected dynamic obstacles. This method was validated by conducting various experiments in maze like test-beds with hallways, a hallway with obstacles, a dead end room, unexpected falling objects, and a dynamic goal leading the robot to a fire.

2.3 Weighted Vector Method (WVM)

2.3.1 Configuration Space

2.3.1.1 Vector Representation

WVM groups vectors to represent obstacles with angles and magnitudes (i.e. distances) which is different than existing methods of obstacle representation that describe obstacles as simple circles or polygons [36-38, 47]. This vector grouping enables the robot to detect various types of obstacles in an environment. As the number of vectors increases, the obstacle shape recognition becomes more accurate. For example, by using Hokuyo Lidar, the surrounding area can be represented by 961 vectors within a 240 degree field of view (FOV). Figure 4 shows how the WVM recognizes obstacles such as general obstacles, maze-like layouts, hallways and corners. In addition, obstacle avoidance methods with sensor-based algorithms [53] require an additional stage to convert the 1-D Polar coordinates sensor data to *x* and *y* on 2-D Cartesian coordinates for the obstacle representation. If a large number of obstacles are present, a higher number of conversions must be processed which leads to slower system processing. In contrast, WVM uses the original sensor data directly without data conversion, making the process faster than FGM.



(a) General (b) maze-like (c) hallway (d) corner

Figure 4. Examples of vector representation for (a) general, (b) maze-like, (c) hallway and (d) corner configurations.

2.3.2 Area of Interest

The higher number of vectors used in the vector representation, the more specific and accurate recognition becomes; however, this causes computational burden. To solve this, the configuration space is divided into four major parts on the basis of interest level. As shown in Figure 5, the parts include the area of protection (AP), area of interest one (AI-1), area of interest two (AI-2), and named area of interest (NAI). AP refers to the space that is within 0.1 m in front of the robot, which is the region where the robot does not have time to avoid an object. A hand or any object placed in this space causes the robot to stop movement immediately as a safety mechanism. Two different areas, AI-1 and AI-2, are divided on the basis of robot speed and processing time. NAI refers to the space within AI-1 that is twice the width of the robot and is adjustable.

The configuration space is divided to increase efficiency during obstacle avoidance by concentrating in areas where obstacles must be avoided and ignoring areas that are irrelevant. For instance, NAI contains information that immediately affects the robot and is treated as first priority in order to avoid collision. AI-1 contains information that influences the robot's heading direction. Objects in AI-2 are irrelevant to obstacle avoidance and does not interfere with navigation, so all information of AI-2 is disregarded. By distinguishing the important obstacles from the unnecessary ones, only relevant information is used, thus reducing the computational burden for obstacle avoidance.

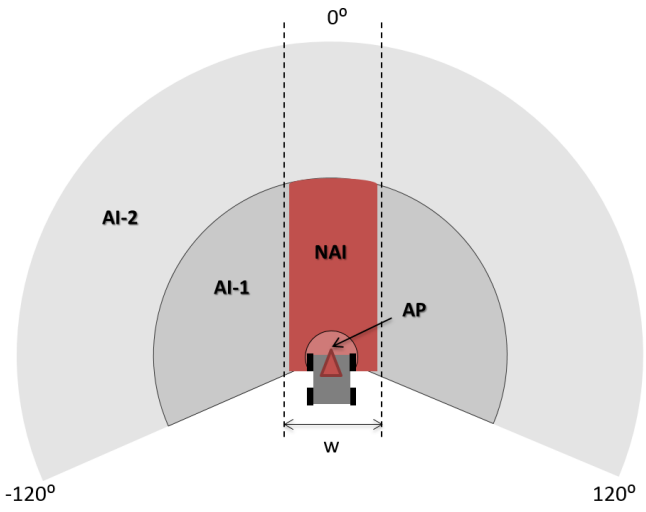


Figure 5. The configuration space is divided into AP, NAI, AI-1 and AI-2 on the basis of the distance from the robot and width of the robot (w).

2.3.3 Goal and Sub-Goal

WVM alternates between goal and sub-goal, depending on obstacle layout, in order to solve the inherent problem of local obstacle avoidance [54]. The goal is a global goal that refers to the location where the robot is trying to go. One example of this is a robot trying to find a fire outside its FOV. When trying to find a fire outside the robot FOV, the goal is dynamic and based on conditions present in the FOV [23, 27]. Recently, a method was developed to provide a heading toward the fire through processing of thermal images [24]. This estimated direction becomes the goal in the WVM. The sub-goal refers to a local goal that is calculated only when obstacles are within NAI and AI-1. The sub-goal becomes the priority when obstacles are in NAI or AI-1, while the goal becomes priority when obstacles are not in NAI and AI-1. This is done so the robot avoids close obstacles which are higher priority than moving toward the goal. In addition, if there are multiple sub-goals, the sub-goal closest to the goal is chosen. The usage of goal and sub-goals allows the robot to avoid local obstacles while moving toward the goal and prevents the robot from wandering aimlessly or losing its global goal, which are some of the problems of local obstacle avoidance [54].

2.4 Decision Tree

Five modes are classified to compute the safe heading of the robot. The modes include hallway, corner, obstacle and trap for the sub-goal. Otherwise, the robot defaults to following the goal. As shown in Figure 4, these modes are determined by using a decision tree with local environment analysis based on the deployment of obstacles in the configuration space. The goal following mode is activated when there are no obstacles to avoid within AI-1. The other modes are identified depending on the location of the obstacle on NAI and AI-1, with consideration of robot width (w) and gap (G) between obstacles. The WVM first checks whether there are obstacles within AI-1. If there are none, the goal following mode is activated which is the priority when there are no obstacles to avoid. When there are obstacles detected in AI-1 but not

in NAI, WVM checks to see whether the obstacles detected are on the left (L), right (R), or both sides (L & R). If the obstacles are on both sides, the mode switches to hallway mode. When the obstacle is on either the left or right, the obstacle is viewed as an object that will not interfere with navigation, and the goal following mode is activated.

When there is an obstacle detected in the NAI, the WVM checks to determine whether there is an obstacle only in NAI or in AI-1 as well. If the obstacle is only in NAI, it checks to see if the obstacle is in front, left (L), right (R), or on both sides (L & R) to determine whether the mode is obstacle, trap, or hallway. If there is enough space for the robot to pass through (e.g. G > w), the hallway mode is activated, but if the passage is not wide enough (e.g. G ≤ w), the trap mode is activated. In the trap mode, the robot turns around to find another passage and if this is not possible, it goes back to its original starting point. In all other circumstances, the obstacle mode will be activated. If the obstacle is on both NAI and AI-1, it checks to find whether the obstacle in AI-1 is just on one side or both sides. If it is on one side only, the corner mode will be activated; however, if on both sides, the WVM will check whether the passage is wide enough for the robot to pass and either trap mode or hallway mode will be activated.

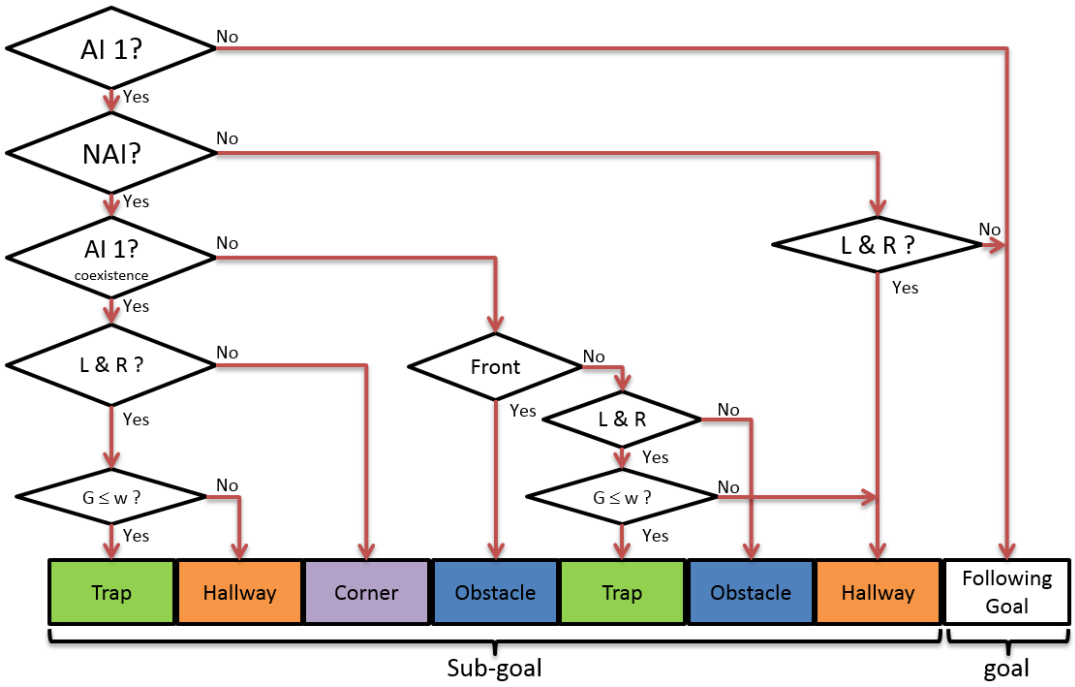
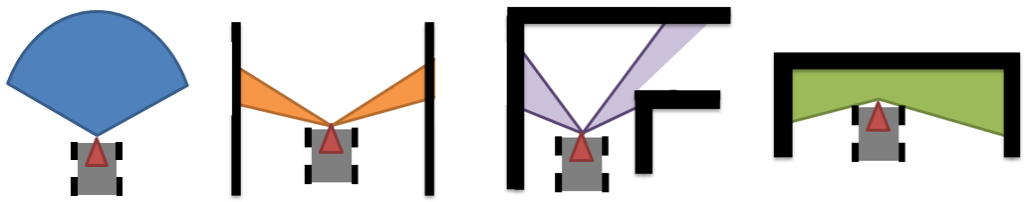


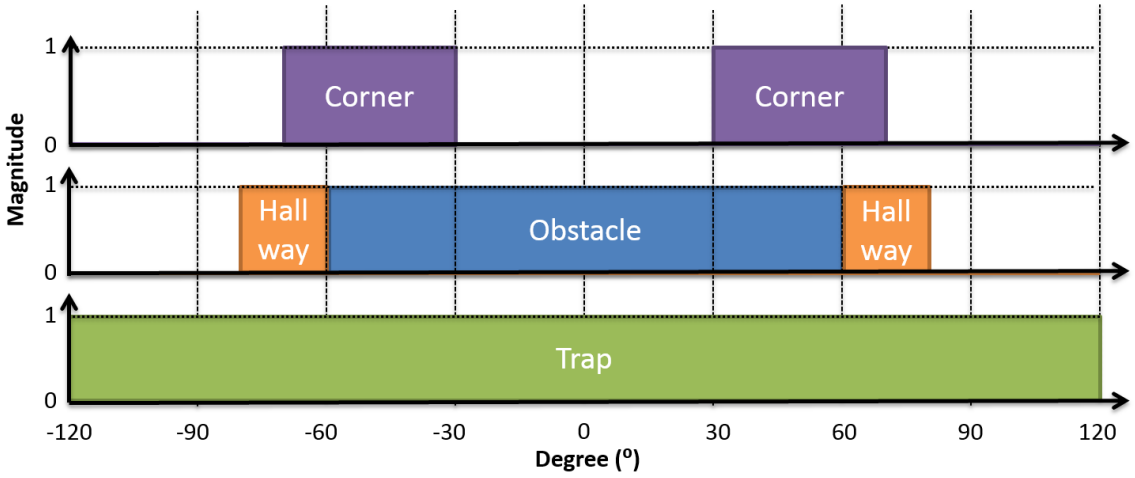
Figure 6. The decision tree used to determine whether the robot follows the goal or applies one of the four sub-goal modes.

2.5 Safe Heading Calculation

Previous algorithms mainly focused on avoiding collisions [55-58]; however, this paper focuses on calculating a safe heading in addition to collision free navigation. WVM perceives the local environment through vectors and uses the decision tree to decide a suitable obstacle avoidance sub-goal mode while computing a safe heading angle. To increase computation efficiency in calculating the safe heading, WVM disregards the unnecessary vectors by allocating weighting factors depending on the sub-goal mode as shown in Figure 7. By varying the weighting factors with angle from the middle of the robot, WVM is able to use the necessary obstacle vectors in calculation of the obstacle avoidance while ignoring other vectors. This leads to fast processing speed that also allows for the recognition of sudden, dynamic obstacles as well as avoidance.



(a) Obstacle (b) hallway (c) corner (d) trap

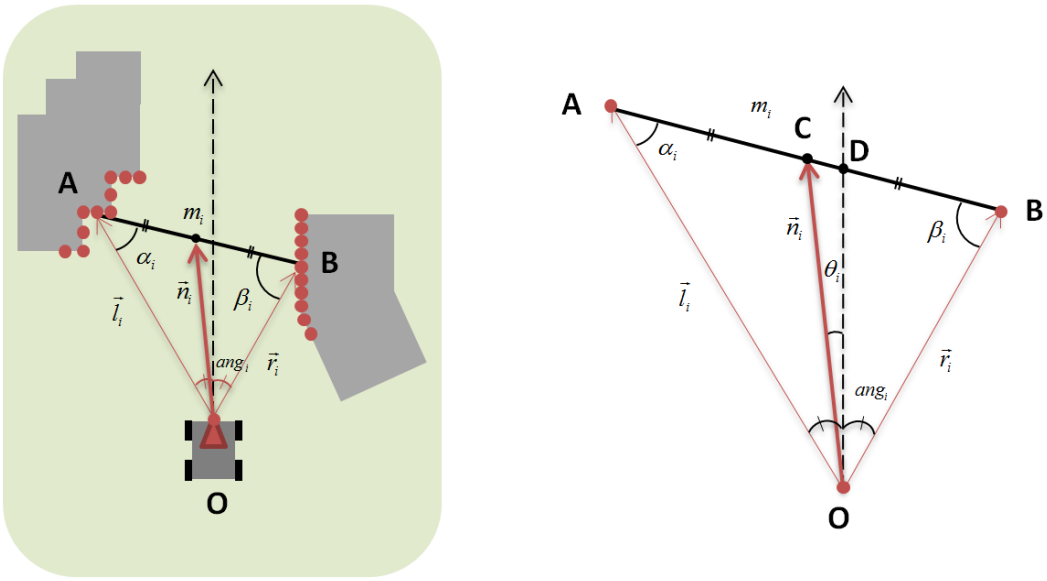


(e) Weighting ranges of the four sub-goal modes

Figure 7. The weighting ranges of obstacle, hallway, corner and trap sub-goal modes.

2.5.1 Hallway and Corner Modes

In fire situations where there is constant risk of falling debris, the safest path for navigation is midway between walls since this would be the furthest point away from the possibly weakening structure [36]. For safe heading calculations, mathematical equations introduced in FGM were applied to WVM. However, because FGM only represents obstacles in the form of circles, there were limitations for practical applications where obstacles had various shapes [15]. In addition, FGM has difficulty in recognizing continuous obstacles such as hallways and corners. By using a vector representation to recognize obstacles, WVM was able to overcome these limitations.



(a) (b)

Figure 8. A basic geometric concept of WVM.

Figure 8 shows a basic concept of WVM. When there are obstacles on both sides of AI-1, WVM identifies all the information generated from a sensor (e.g. Lidar) as vectors and takes pairs of vectors that match each other symmetrically about the robot heading direction (which is 0^0 based on the robot frame) to calculate the robot path. Red dots in Figure 8 refer to vectors that outline the sides of obstacles. A pair of vectors among them (i.e. $\vec{l}_i$ and $\vec{r}_i$) is selected to compute

the direction the robot needs to move as shown in Figure 8(b). The distance between obstacles is found by

$$m_i = \sqrt{\left|\vec{l}_i\right|^2 + \left|\vec{r}_i\right|^2 - 2\left|\vec{l}_i\right|\left|\vec{r}_i\right|\cos(2 \cdot ang_i)} \tag{1}$$

Since $\vec{l_i}$ and $\vec{r_i}$ are symmetrical vectors about the robot heading direction with an angle of ang_i, $|n_i|$, linked to velocity of the robot, can be calculated by Apollonius theorem,

$$|n_i| = \frac{1}{2}\sqrt{\left|\vec{l}_i\right|^2 + \left|\vec{r}_i\right|^2 + 2\left|\vec{l}_i\right|\left|\vec{r}_i\right|\cos(2 \cdot ang_i)} \tag{2}$$

Angles of α_i and β_i, linked to the robot turning, are calculated by

$$\alpha_i = \cos^{-1}\left(\frac{\left|\vec{l}_i\right|^2 + m_i^2 - \left|\vec{r}_i\right|^2}{2\left|\vec{l}_i\right|m_i}\right) \tag{3}$$

$$\beta_i = \cos^{-1}\left(\frac{\left|\vec{r}_i\right|^2 + m_i^2 - \left|\vec{l}_i\right|^2}{2\left|\vec{r}_i\right|m_i}\right) \tag{4}$$

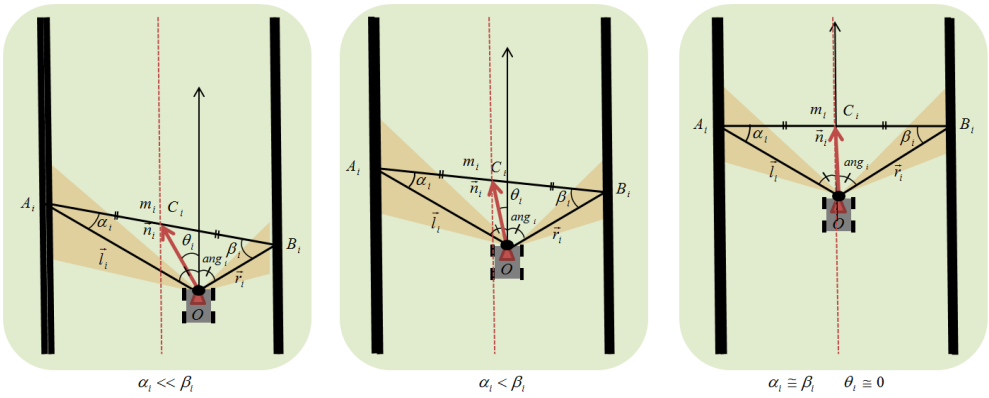
The direction between the direction the robot is currently heading and the angle it needs to turn, θ_i, is

$$\theta_i = \begin{cases} \left|\cos^{-1}\left(\dfrac{\left|\vec{l}_i\right|^2 + \left|\vec{n}_i\right|^2 - (0.5m_i)^2}{2\left|\vec{l}_i\right|\left|\vec{n}_i\right|}\right) - ang_i\right| & \text{if } \beta_i \langle \alpha_i, \\ -\left|\cos^{-1}\left(\dfrac{\left|\vec{l}_i\right|^2 + \left|\vec{n}_i\right|^2 - (0.5m_i)^2}{2\left|\vec{l}_i\right|\left|\vec{n}_i\right|}\right) - ang_i\right| & \text{otherwise} \end{cases} \tag{5}$$

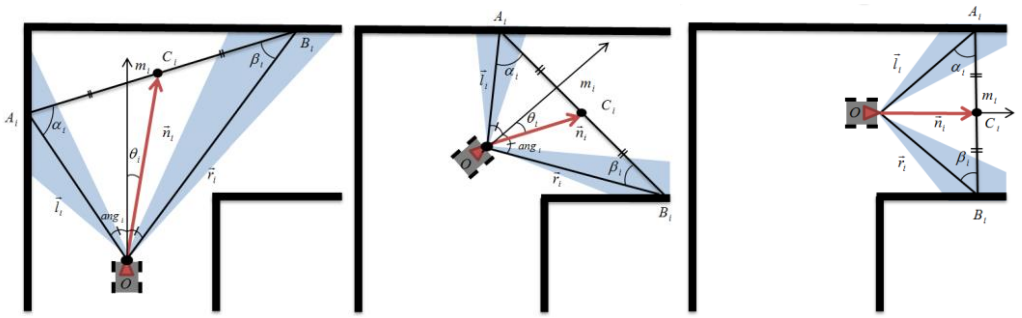
The difference between α_i and β_i is related to the direction in WVM. When α_i is small compared to β_i, θ_i takes on a negative value which makes the robot turn left. On the contrary, when α_i is larger than β_i, θ_i takes on a positive value which makes the robot turn right. Since the magnitude difference between α_i and β_i is proportional to the magnitude of the direction $|\theta_i|$, the robot makes a wider turn when the difference between two angles increases. In addition, when α_i and β_i are equal, it means that the robot is midway between obstacles which is the position WVM leads the robot by continuously updating the value of θ_i. As previously explained, WVM performs calculations using the non-zero vectors that result from applying the appropriate weighting functions. Using Equation (5), the sub-goal can be calculated as,

$$\theta_{sub-goal} = \frac{1}{p}\sum_{i=1}^{q}(\theta_i), \tag{6}$$

where p is the number of weighted vectors and q is the number of vectors from a sensor. In addition, the sub-goal is calculated by the sum of directions then divided by the number of weighted vectors. Equation (6) is continuously applied to compute the heading during the hallway and corner modes to safely navigate the robot. For example, Figure 7 shows how a hallway and corner is navigated.



(a) hallway



(b) corner

Figure 9. A graphical representation showing how the robot navigates (a) a hallway and (b) corner geometry.

2.5.2 Obstacle

This mode is activated when an obstacle or unexpected dynamic obstacle (e.g. falling debris) appears within NAI. This mode occurs less frequently than the hallway and corner modes since obstacles have already been avoided by the other modes. In addition, this mode has the highest priority of all five modes because it is most urgent to avoid the obstacle that is blocking the path directly in front of the robot. The sub-goal is calculated depending on whether the obstacle is on the left or right side in the NAI using Equation (7) or (8), respectively. If the obstacle appears on the left of NAI, WVM calculates the horizontal distance between the obstacle and the robot finds the vector with the shortest horizontal distance and uses the vector angle to calculate the sub-goal. If the horizontal distance is greater than the width of the robot, $\left|\vec{l}_i sin\theta_i\right| > w$, which means there is no obstacle within NAI, $\theta_{sub\text{-}goal}$ becomes zero.

$$\theta_{sub-goal} = \begin{cases} 90 + \underset{\theta_i \in (-120,0)}{\arg\min}\left(2\left\|\vec{l}_i\right|\sin\theta_i\right|\right) & \min\left(\left\|\vec{l}_i\right|\sin\theta_i\right|\right) \le w \\ 0 & otherwise \end{cases} \tag{7}$$

$$\theta_{sub-goal} = \begin{cases} \underset{\theta_i \in (0,120)}{\arg\min}\left(2\left|\vec{r}_i\right|\sin\theta_i\right) - 90 & \min\left(\left|\vec{r}_i\right|\sin\theta_i\right) \le w \\ 0 & otherwise \end{cases} \tag{8}$$

Figure 10 shows an object located on the left within NAI. WVM finds the vector with the shortest horizontal distance and uses the vector angle to calculate the sub-goal. By updating the sub-goal, the robot turns to the opposite direction until the obstacle is no longer visible in NAI. The robot then moves forward until it passes the obstacle, then immediately the goal following mode is activated again and the robot turns to face the goal and moves forward.

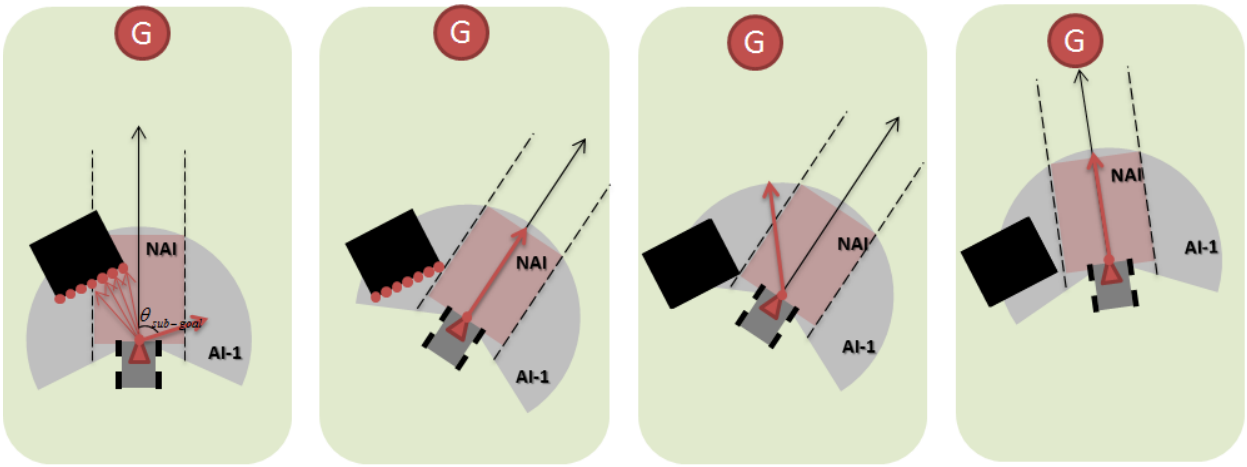


Figure 10. A graphical representation showing the robot movement in obstacle mode.

2.5.3 Trap

This mode is activated when the robot encounters a dead end or when there are obstacles in NAI and the open passages are not wide enough for the robot to pass in between. In this mode, the robot turns around to find another passage and if this is not possible, it goes back to its original starting point. The weighting range of this mode includes all vectors in the FOV (i.e. −120° ~ +120°). WVM finds a vector angle that has the maximum magnitude and the angle becomes the sub-goal,

$$\theta_{sub-goal} = \underset{\theta \in (-120,120), i \in (1,961)}{\arg\max} \left(mag_i(\theta_i) \right), \tag{9}$$

2.6 Experiments and Results

2.6.1 Mobile Robot Platform

A four-wheeled skid-steering style robot was used to demonstrate the WVM in different indoor environments. The robot was 0.54 m long and 0.56 m wide. Four high torque DC motors were installed to support its maneuverability using various sensors. For autonomous control over the DC motors, a microcontroller (Arduino Uno) and a servo controller were used which allowed maximum speed of the robot, 40 cm/sec. To reduce the payload, the frame was made of aluminum. The platform was equipped with a 24V battery to power the DC motors and three 12V batteries for the sensors.

The mobile robot platform [24] in Figure 11 was equipped with a fit-PC and several sensors such as a web camera, LWIR camera, LIDAR, and UV sensor. The fit-PC included an i7 Intel core processor and Wi-Fi 802.11b/g/n. This on-board computer connected the sensors and handled the data processing. Labview was installed on this computer and was used to acquire data from the sensors and display visual and thermal images. In addition, a Labview interface for the Arduino (LIFA) was employed to communicate between the Arduino and algorithms developed for the robot motion control. A web camera was mounted on the top of the robot. To provide constant visual images and visual record of each experiment, a FLIR Tau 2 320 long wavelength (7-14μm) IR (LWIR, $49^{\circ} \times 39^{\circ}$ FOV), camera was installed beside the web camera. It produced a thermal image stream with a resolution of 640×480 which was down-sampled to 360×240 as the images were acquired. The thermal images were used in the seek-and-find algorithm to calculate the heading toward a fire outside the robot FOV, which is the goal of one of the indoor tests in this paper. A LIDAR, Hokuyo UTM-30LX, was mounted at the front of the robot, which produced vectors at angles between -120 and 120 degrees over a horizontal plane. The UV sensor, Hamamatsu C10423 wavelength of 200nm, was located between the visual and LWIR cameras, and was used by the Seek-and-Find algorithm to discriminate fires from hot

objects. The UV sensor field of view was narrowed to 20 degrees by optimizing its focal length to be more consistent with the LWIR camera.

The orientation plays a major role in controlling the motion of a differential drive mobile robot [59, 60] such as turning left/right, moving forward/backward and stop. The robot motion was performed by a *turn-then-travel* strategy [60], which was controlled by the heading direction that is frequently updated during navigation. Different control parameters per direction are computed and used for robot motion control. This repetitive computation is not efficient during real-time implementation. A look-up table [61] in matrix form was used instead to store large numbers of control parameters per direction, optimized from experiments. By retrieving existing information from memory instead of undergoing extensive computation, the robot can be controlled faster. The total processing time for obstacle avoidance decision was 0.2 s, which included the robot recognizing an obstacle, deciding a mode through the decision tree, calculating safe trajectory, and taking a physical action. This response time is fast enough to avoid sudden, dynamic obstacles that appear in front of the robot in real time.

The robot moves forward when θ_{final} and the robot heading direction match; however, when the matching angles were too narrow, oscillation can occur. In order to avoid this problem, the angle range for forward movement was extended to $\pm 5°$.

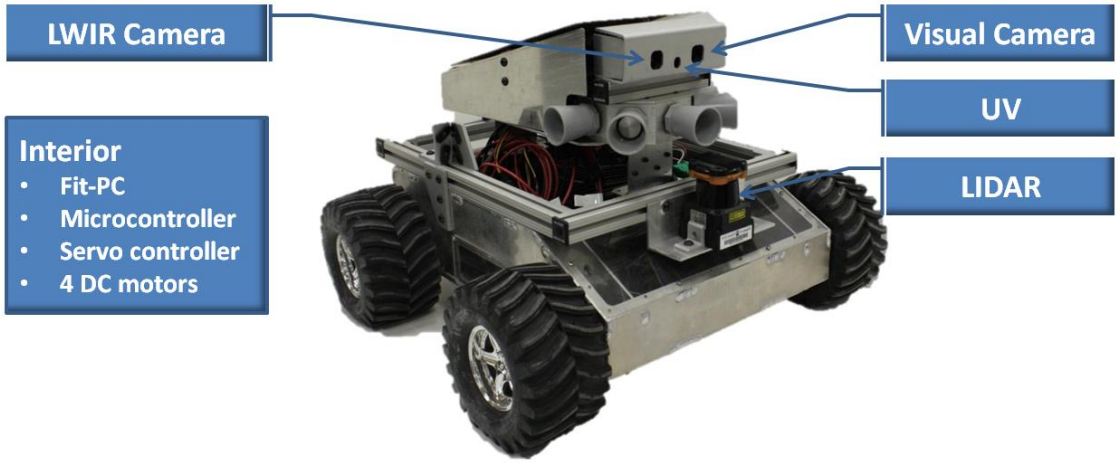


Figure 11. The mobile robot platform sensor layout.

2.6.2 Test Environments and Results

The test-beds were designed to demonstrate various situations commonly encountered for indoor environments including a hallway and corner maze (hallway with changing direction), hallway with obstacles, dead end room, and unexpected dynamic obstacles to represent falling items. All of these environments were used to evaluate the ability of the WVM to perform obstacle avoidance. WVM continuously switched between the five modes depending on the situation at hand until the robot reaches its final goal. In addition, an environment with a dynamic goal for finding a fire was used along with the Seek-and-Find algorithm to demonstrate the ability of the algorithm to switch between the sub-goal and the goal in order to reach the fire.

2.6.2.1 Hallway and Corner Maze

The test environment in Figure 12 included continuous obstacles in the form of a maze of hallways and connecting corners. With walls 1.3 m apart, the total travel distance through the maze was 7.4 m. Figure 12 contains a plan view of the setup as well as the approximate path of the robot. As shown in Figure 13, the robot starts outside the maze and immediately finds a safe heading into the maze. The robot moves through the hallway taking a total of 3 left turns and 2 right turns. At the end, the robot moves out through the middle of the exit. The navigation through the maze required continuous switching between hallway and corner modes.

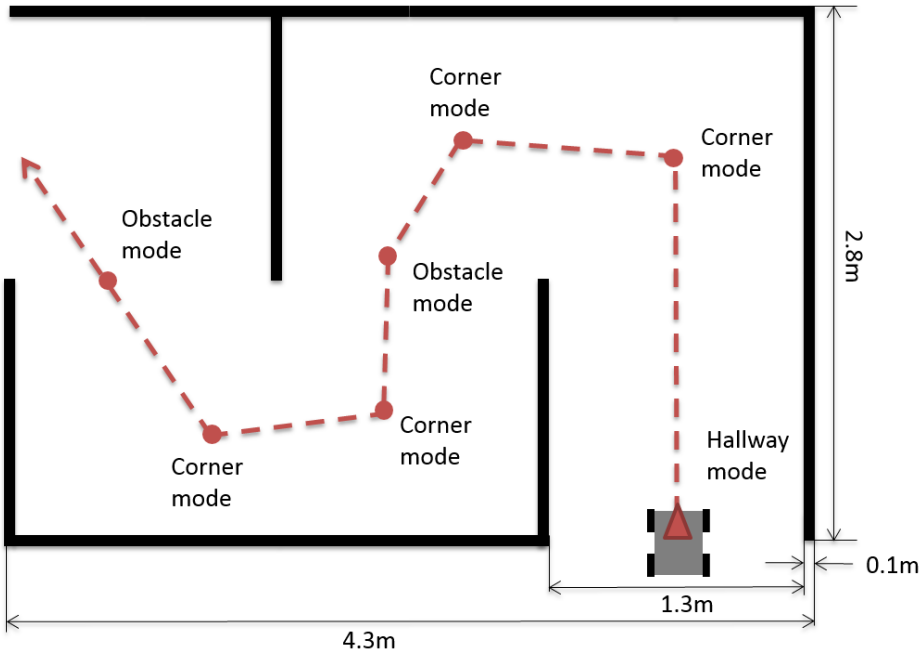
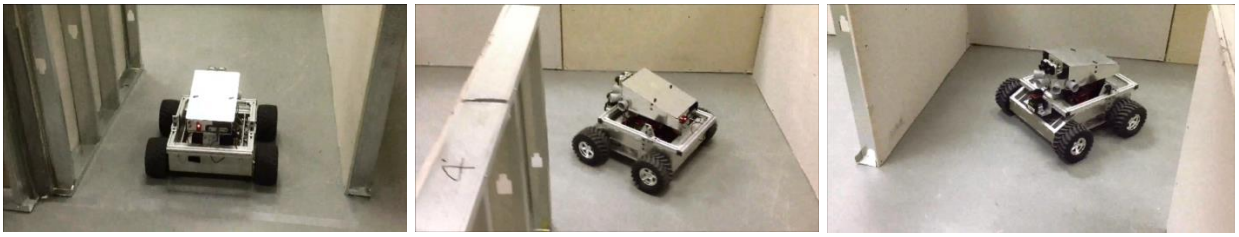


Figure 12. The hallway and corner maze and a plan view showing a sketch of the robot path.



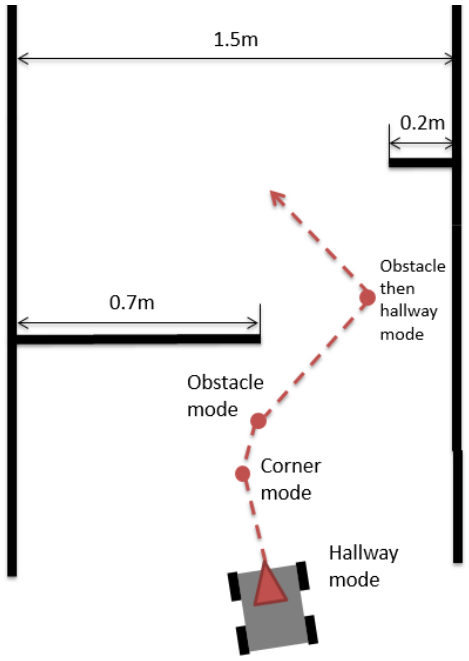
(a) hallway mode (b) corner mode (c) corner mode

(d) obstacle mode (e) corner mode (f) obstacle mode

Figure 13. Experiment results in the environment with a maze of hallways and corners.

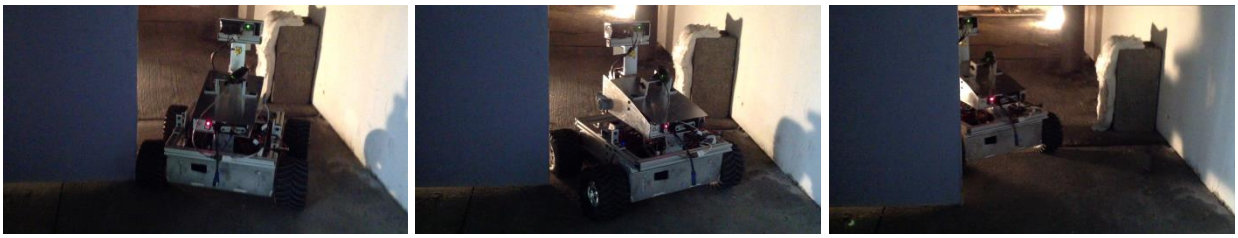
2.6.2.2 Hallway with Obstacles

This environment was designed to have a maze-like layout with discrete obstacles in the hallway. As seen in Figure 14, the hallway was 1.5 m wide with a 0.7 m wide, 1.2 m high obstacle on one side and a 0.2 m wide, 0.4 m high concrete block on the opposite side. The closely placed obstacles required the robot to take a very wide turn while finding the safe heading into the narrow passage. The gap between the plate and the wall was 0.8 m and the robot width was 0.56 m. Despite the similar widths, WVM was able to overcome the obstacles by quickly and continuously switching modes. As shown in Figure 14, at the entrance of the environment, the hallway mode activates and the robot starts its navigation. As the robot moves in, it recognizes the plate on the left and switches to corner mode and turns right. Because the plate comes within the NAI range during the turn, obstacle mode is activated and the robot turns further to the right until the plate is no longer within NAI. As this point, the robot immediately starts moving forward. As the robot moves forward, it encounters the concrete block on the right and switches to obstacle mode, turns left, then comes back to hallway mode after passing the obstacle.



(a) plan view showing a sketch of the robot path

(b) hallway mode (c) corner mode (d) obstacle mode

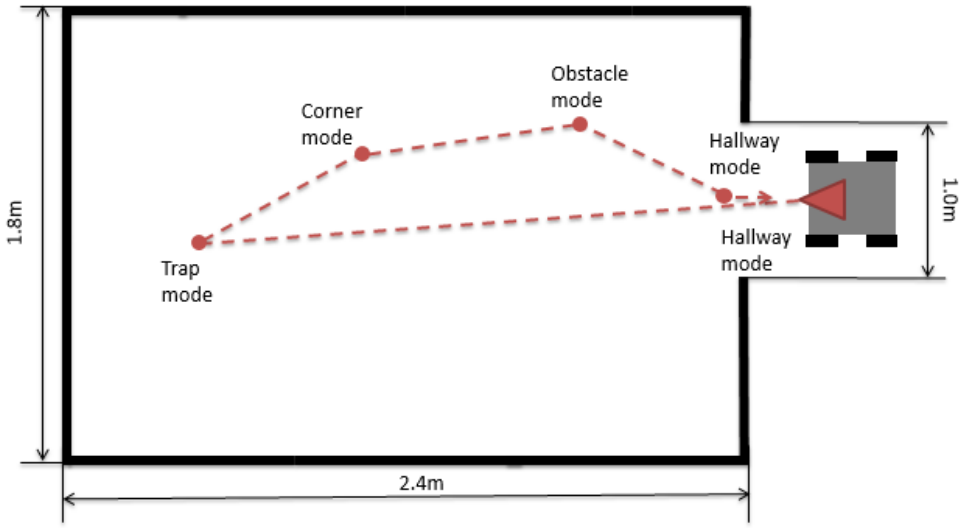


(e) moving forward (f) obstacle mode (g) hallway mode

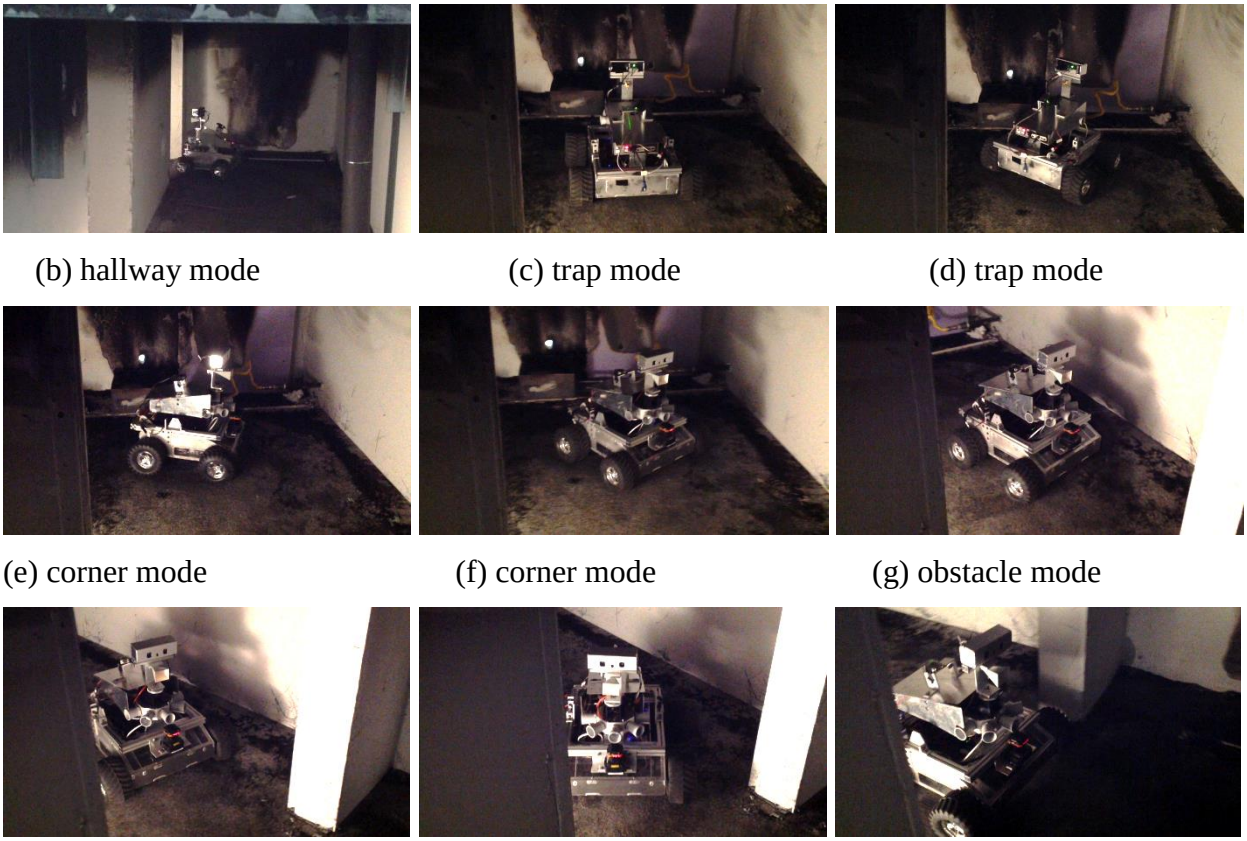
Figure 14. Navigation through a hallway with various obstacles on both sides.

2.6.2.3 Dead End Room

An environment with a room having a single door was used to demonstrate how the WVM is capable of overcoming a dead end room. The room was 2.4 m long 1.8 m wide and 2.0 m high with opening of 1.0 m wide, 1.8 m high. As shown in Figure 15, the robot was started just outside the room and hallway mode was activated for the robot to safely enter the room. When the robot reached the end of the room, trap mode then corner mode was activated making the robot turn right. As the obstacle moved out of the NAI, hallway mode was activated and the robot moved toward the door. When the robot reached the opening, obstacle mode was again activated to avoid the side of the wall but immediately switched to hallway mode to safely pass through the narrow opening.



(a) plan view showing a sketch of the robot path



(b) hallway mode (c) trap mode (d) trap mode

(e) corner mode (f) corner mode (g) obstacle mode

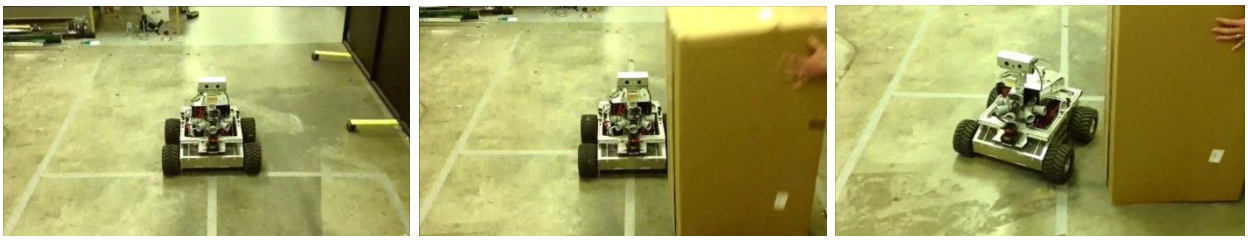
(h) hallway mode (i) obstacle mode (j) hallway mode

Figure 15. Navigation out of a dead end room with one door opening.

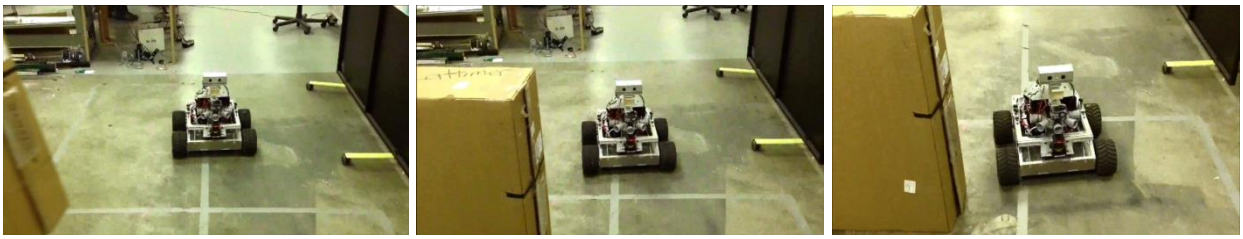
2.6.2.4 Dynamic Obstacles

The robot performance when unexpected dynamic obstacles become present was evaluated in an open environment where a box was suddenly placed in front of the robot. The box was 0.34 m long, 0.60 m wide and 0.96 m high. Experiments were conducted by suddenly placing the box within and slightly outside of the NAI. As shown in Figure 16, when the box suddenly appears in front of the robot to the right in the NAI, the robot avoids the dynamic obstacle by immediately stopping and turning until the obstacle is no longer within NAI. When the box is placed on the left side of NAI, again the robot is shown in Figure 16(b) to avoid the obstacle by stopping and turning to the right. In the case where the box suddenly appears right outside of NAI, the robot calculates that it does not need to avoid it and keeps moving forward as seen in Figure 16(c). Lastly, when the unexpected obstacle suddenly appeared only partially within NAI, the robot turned slightly to avoid the obstacle, as shown in Figure 16(d).

(a) Unexpected dynamic obstacle on right side in NAI



(b) Unexpected dynamic obstacle on left side in NAI



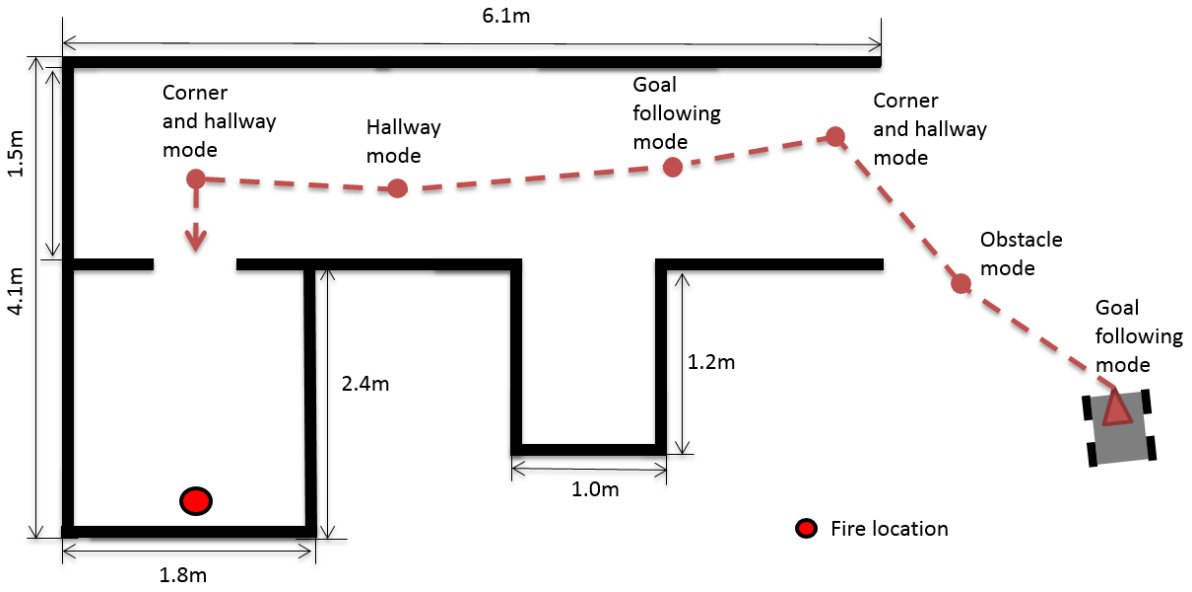
(c) Unexpected dynamic obstacle outside NAI

(d) Unexpected dynamic obstacle partially within NAI

Figure 16. Unexpected dynamic obstacle avoidance.

2.6.2.5 Dynamic Goal for Finding Fire

WVM was evaluated in a simple indoor environment where a hallway with two rooms were present, with one of the rooms containing a fire using a propane sand burner [24]. WVM dynamically used the goal or sub-goal depending on the surrounding area. The goal was calculated by the 'Seek-and-Find' algorithm that provides information regarding the robot heading angle using the LWIR image as well as the existence of a fire in the FOV [24]. The sub-goal was computed by WVM to provide safe heading for obstacle avoidance. Due to the uncertainty of the fire location, the heading that is consistently updated by the 'Seek-and-Find' algorithm becomes the goal shown as a red vertical line in Figure 17. However, because the updated goal heading is relative to the robot orientation, whenever the robot moves or avoids obstacles, the goal dynamically changes. Therefore, the goal or sub-goal is accordingly applied by the WVM to avoid obstacles and update the heading angle in real time in order to ultimately reach the final goal (i.e., fire).



(a) Plan view showing a sketch of the robot path

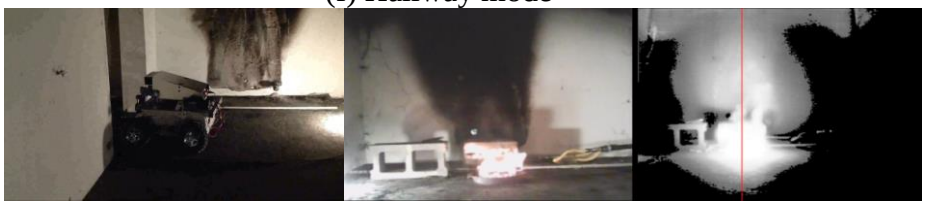
(b) Goal following mode

(c) Obstacle mode

(d) Corner and hallway mode

(e) Goal following mode

(f) Hallway mode



(g) Corner and hallway mode

Figure 17. Navigation with a dynamic goal for finding a fire. Left, center and right images display the robot motion, visual image and thermal image from the robot, respectively. The red vertical line in the right images represents the heading to the goal calculated by the Seek-and-Find algorithm.

2.7 Conclusion

This paper has presented the weighted vector method (WVM) as a real-time local obstacle avoidance method for autonomous robots in indoor environments where there may be dynamic obstacles and goals. WVM uses geometrical concepts and formulations of FGM and VDF for safety trajectories and fast motion response in dynamic environments. However, due to limitations in obstacle recognition, those methods cannot be reliable for indoor environments where cluttered, complex and various shapes of obstacles are present. To resolve this, WVM uses the vector representation to perceive obstacles of various shapes. To reduce computation time to allow real-time implementation, WVM divides the configuration space into 4 areas and distinguishes necessary and unnecessary obstacle information in local environments. Based on deployment information of necessary obstacles, WVM uses five modes that are determined by a decision tree. Each mode has different weighting factors for obstacle avoidance. The weighted vectors in each mode are used to calculate safe headings for robot movement. Furthermore, WVM integrates goals and sub-goals for obstacle avoidance and overcomes problems in local obstacle avoidance. The WVM was implemented onto a mobile robot platform and demonstrated to successfully navigate through different test environments by sequentially switching modes to overcome obstacles in the surroundings, including avoiding dynamic obstacles.

3 Local Dynamic Goal-Based Fire Location

3.1 Abstract

Finding a fire fast is crucial in firefighting. For risky situations, it would be ideal to send a firefighting robot into the structure that could quickly and efficiently find the fire and suppress it. This paper introduces an algorithm developed for an intelligent firefighting mobile robot to find a fire efficiently by fusing long wave infrared camera, ultraviolet radiation sensor, and LIDAR. For its validation, an experimental test-bed was constructed with a hallway and two rooms, with one of the rooms containing a real size fire created by propane gas. The robot immediately calculates its path towards the fire, moves towards it avoiding obstacles, and ultimately finds the fire. When the fire is out, the robot returns to its original starting place.

3.2 Introduction

Firefighting is a difficult and dangerous job that puts fire fighters at risk. According to the U.S. Fire Administration (USFA), a provisional total fatality of on-duty firefighters was 83 for the years 2011 and 2012 [1, 2, 4]. In order to reduce losses of fire fighters and increase their effectiveness on the job, firefighting robots have been extensively researched. The need for efficient firefighting robots has been presented on other countries as well. For example, Japan developed Rainbow 5 [17], an unmanned robot that has the capability to handle big fires such as petrochemical complex fires, aircraft crash fires, etc. This robot is equipped with an obstacle remover to discard fallen or dangerous objects. Another robot is LUF60 [13], designed in Germany, this was created to exhaust gas and extinguish fires from sixty meters away. F2GV [13], designed in Malaysia has a track system with heavy rubber track belts. It also has maximum speed of 2.36 km/h and curb weight of 910 kg. A water hose mounted on the top of the robot is remotely controlled to suppress fire. Unfortunately, these robots were designed to perform from an outdoor location; they are too big to be utilized indoors.

There are several firefighting robots applicable to indoor tasks. In South Korea, Fire Searcher [12], a scout robot is sent to hazardous sites where fire and poisonous gas exist. This robot delivers internal conditions about fire and victims of the building it is in. The operator can also directly communicate with the trapped victims via wireless communication, allowing for an efficient rescue plan. Tehzeeb [21] is a semi-autonomous robot designed to rescue victims. This robot can localize itself and build a map of its surrounding as it searches. Another indoor firefighting robot is Autonomous Fire Fighting Mobile Platform [14] which was designed to detect flame and extinguish it by using a fan mounted on top. A set of black lines guides its patrolling movement with line following techniques. Another from Korea is the portable fire evacuation guide robot [18] which can be remotely controlled and was built to be thrown into high risk areas and transfer environmental information back to the operators. In addition, it is capable of searching for victims and helping them evacuate by voice communication between firefighters and the victims themselves.

All the above mentioned indoor robots have the capability of performing firefighting tasks indoors and some even in tight spaces due to their smaller sizing. However, some are nonautonomous which makes them mainly dependent on a remote operator to decide where to move and search. In addition, limitations of the current wireless communication technology can create unstable connections that make data collection and communication challenging. These robots also only allow a small and narrow view for the operator which may make it difficult for the operator to find fire and make quick decisions. Although some of the robots do have autonomous systems to do their own tasks, they might spend much time moving restlessly in areas where there is no fire before finding fire. All in all, these robots' functions may waste crucial time during urgent situations.

In this paper, we introduce the seek-and-find fire algorithm that enables a firefighting robot to reach fire fast by analyzing temperature distribution of its surrounding area. It also calculates and determines the best way for the robot to approach fire by avoiding obstacles. Thus, the proposed algorithm is able to increase effectiveness of task in firefighting. The algorithm was also developed for SAFFiR (Shipboard Autonomous Firefighting Robot), an autonomous humanoid firefighter, but to validate the proposed algorithm, it was first tested on a four-wheel skid-steering autonomous mobile robot platform.

3.3 System Architecture

This section describes the system architecture of the firefighting robot shown in Fig. 1. The hardware of this system includes an autonomous mobile robot platform, tablet PC for supervision, and sensors such as long wave IR (LWIR) camera, LIDAR, ultraviolet radiation (UV) sensor, and a visual camera. Each sensor provides information about temperature, UV level, and distances to objects within its field of view. Together with the three different modes (obstacle, fire locating, and near-fire modes), the seek-and-find algorithm synthesizes all the information gathered to determine the most efficient behavior the robot needs to undertake to move and locate the fire.

The robot is sent in difficult indoor fires where it may be too dangerous for fire-fighters to perform. A table PC can be used to oversee the situation from a remote location. The tablet PC operated by iOS can be connected with the computer system inside of the robot to provide visual and thermal information, UV light level, and information about distance for the navigation. Moreover, the robot can be remotely controlled by the firefighter through the tablet PC; however, in the experiments described herein the robot was completely autonomous.

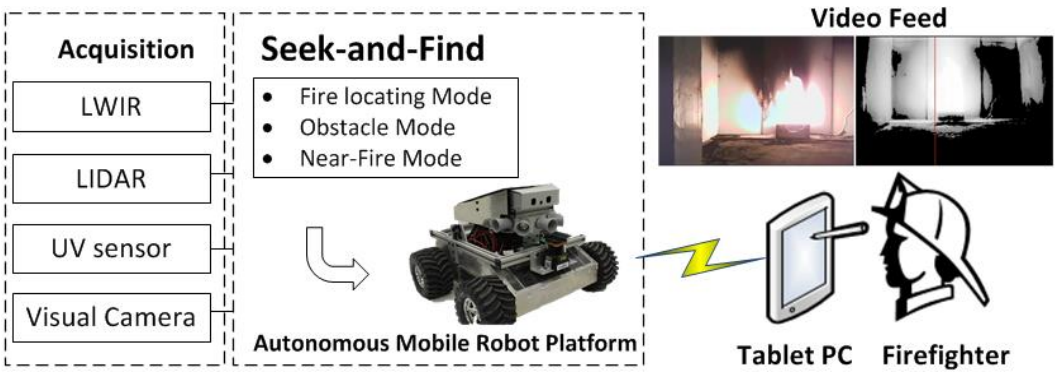


Figure 18. The system architecture

3.3.1 Autonomous Mobile Robot Platform

The autonomous mobile robot platform was built to validate the proposed algorithm before applying it on the humanoid robot SAFFiR. The dimension of the platform was 0.54 meter in length and 0.56 meter in width to accommodate missions indoors where the doors and hallways are narrow. In addition, four high torque DC motors were installed to support various sensors and the suppression equipment for firefighting tasks. For the robot to autonomously control the DC motors, a microcontroller (Arduino Uno) and a servo controller were used which allowed the robot to move up to 10cm/sec. To reduce the payload, the frame was built with aluminum. For a forty minute runtime, the robot was equipped with a 24V battery to power the DC motors and three 12V batteries for the sensors.

3.3.2 Sensors

The robot is equipped with a fit-PC and several sensors such as web camera, LWIR camera, LIDAR, and UV sensor. The fit-PC has an i7 Intel core processor and Wi-Fi 802.11b/g/n. It connects the sensors and handles the data processing. LABVIEW is installed on this computer and is used to acquire data from the sensors and display visual and thermal images. In addition, a LABVIEW interface for the Arduino (LIFA) is employed to communicate between the Arduino and algorithms developed for the robot motion control. A web camera is mounted on the top of the robot. It provides constant visual images and records them during the experiment. A FLIR Tau 2 320 long wave IR (LWIR) camera is installed beside the web camera. It produces a thermal image stream with a resolution of 640×480 which is down-sampled to 360×240 as the images are acquired. The thermal images are turned into the seek-and-find algorithm to calculate the best way to find fire. A LIDAR, Hokuyo UTM-30LX, is mounted in the front of the robot. It measures the distance to objects at angles between -120 and 120 degrees over a horizontal plane. This information is used for navigation. The UV sensor, Hamamatsu C10423, is equipped between the visual and LWIR cameras. Because the UV sensor picks up UV lights from 180 degrees, the field of view was narrowed to 20 degrees by optimizing its focal length.

Table 3. Hardware specification of the mobile robot.

Category	Specification
Microcontroller	Arduino Uno
Servo controller	Sabertooth motor controller
Fit-PC	Intel i7 processor
Laser Range Finder (LIDAR)	Hokuyo UTM-30LX
Visual Image Camera	Logitech HD webcam c270
Long Wave Infrared Camera	FLIR Tau 2
UV sensor	Hamamatsu C10423

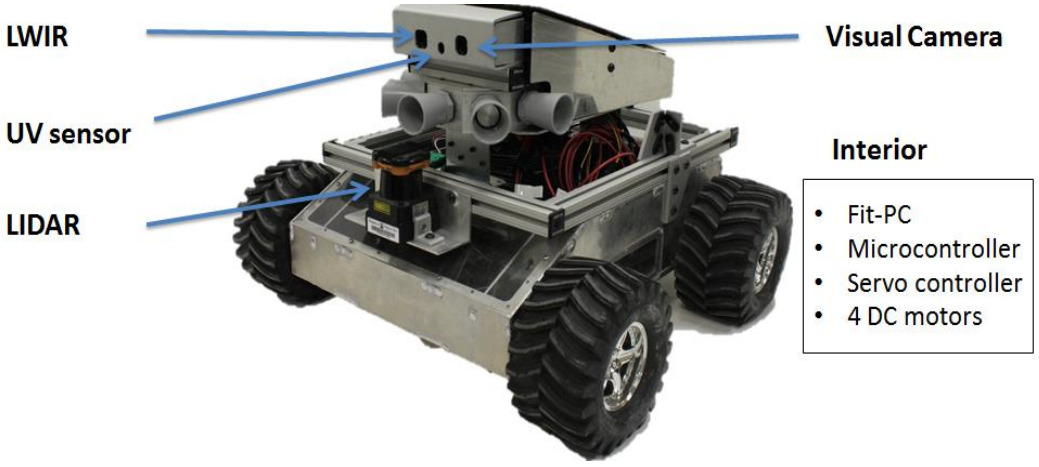


Figure 19. The overall architecture of the autonomous mobile robot platforms.

3.4 The Seek-and-Find Fire Algorithm

To take advantage of the sensors and make the robot able to reach the fire effectively, sensor fusion is applied. The sensor fusion technique is crucial in order for the sensors to work efficiently to achieve the task of finding the fire. There are many different situations the robot could face during its firefighting task. Each situation requires different sensors to be fused. To solve this problem, three modes were defined; fire locating mode, obstacle mode and near-fire mode. Each mode allowed the robot to react to different situations accordingly until ultimately finishing its task.

3.4.1 Fire locating Mode

The fire locating mode activates when there is a fire somewhere in the structure but there are no obstacles within a meter. In this mode, the robot analyzes the relative thermal intensity distribution generated by the LWIR camera. Each pixel of the image includes a relative thermal intensity value expressed as an integer between 0 to 255. This range indicates temperature level with respect to the rest of the scene. For example, if a pixel is 0, it means it has very low intensity. In contrast, if it is 255, it means it has a very high intensity relative to the gain temperature range and other objects in the scene. The maximum intensity tracking algorithm was developed for gathering this temperature-related intensity information from the image and determining which vertical columns or horizontal rows contained the highest intensity pixels. This result is finally translated into a direction for the robot to move, which is also indicated as a vertical red line on the thermal image. However, this algorithm does not consider surface materials, ambient temperature and existence of other secundary radiation sources. The fire locating mode allows the robot to keep following the intensity distribution produced by the fire in an indoor environment. Thus, as long as there are no obstacles, the direction determined by the maximum temperature tracking algorithm works as the highest priority.

3.4.2 Obstacle Mode

The obstacle mode activates when the robot faces obstacles. In this mode, the distance information from LIDAR is a priority over the thermal information. However, the seek-and-find algorithm does not depend solely on LIDAR information. In other words, the LWIR and LIDAR

are fused together in order not to lose its original goal towards the fire by keeping the direction calculated during the fire locating mode. In this mode, the vector field histogram (VFH) [9] is applied at first to identify the robot's possible paths. Then, the most convenient path is selected by integrating the possible paths with the direction generated by the fire locating mode. At the same time, the robot takes its own width into consideration when calculating the gap between obstacles or walls. The robot is also able to calculate the precise medial path it needs to take between obstacles it is about to pass. Additionally, this mode can detect upcoming rooms by detecting slanting on the path. If the slanting is large enough that the robot determines it needs to accommodate itself to be in a new medial space, it will slightly turn its body. But if the robot does not detect a high temperature in the room, it passes it and continues its movement to the initial direction calculated earlier.

3.4.3 Near-Fire Mode

The near-fire mode activates when the UV sensor detects a strong UV light emission. This mode fuses LWIR camera and UV sensor in order to detect fire and face it. The UV sensor outputs signal pulses when it perceives UV light radiated from fire. Because the frequency of these signals is related to the intensity of the incident UV light, the frequency varies up to 40 Hz when the robot is close to a fire. Thus, analyzing the frequency helps the robot judge whether there is fire nearby. Due to the effects of cosmic rays and/or static electricity, the output pulses are infrequently created at a lower rate. Through this characteristic, the UV sensor output was divided into three levels; weak, normal, and strong. The near-fire mode is activated only when strong UV emission is detected. However, the UV sensor can only tell there is fire nearby; it does not provide information of the exact location or direction of the fire. Therefore, the robot is programmed to stop at the moment the UV sensor detects strong UV emission. From there it relies on the maximum temperature tracking algorithm used in the fire locating mode to analyze the direction of the highest temperature and turns its body to face the fire. The robot will stay in this mode until there is no longer a fire. The robot will start moving again when the strong UV light is no longer present. A movement will only be made after determining its new mode; obstacle versus fire locating mode.

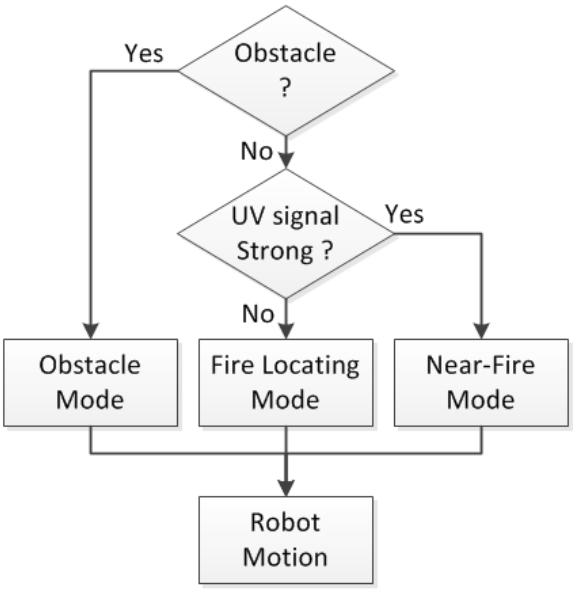


Figure 20. The flow chart of the 'Seek-and-Find' fire algorithm.

3.4.4 Synthesized Algorithm

Figure 20 describes the flow chart of the seek-and-find algorithm. The fire locating mode plays the role of a compass when under an indoor fire where there are walls, rooms, and obstacles to avoid. Because fire influences temperature in the rooms, the robot tracks and searches fire by the maximum intensity tracking algorithm, which is able to analyze thermal intensity distribution from LWIR camera. Because this process is repeated every 0.2 seconds, the robot is able to act according to the different situations the robot encounters along its path. The process starts by determining whether there is an obstacle within a meter distance away, if there is an obstacle, the obstacle mode will be activated but if there isn't, it will move on to the next stage. At this time, the UV sensor output is checked and if it is strong, the near-fire mode will be activated. If the UV sensor signal is low or normal, it will switch to fire locating mode and continue its search. The obstacle mode has a general navigation algorithm of an autonomous mobile robot. This enables the robot to avoid obstacles and plan its path while seeking direction towards the fire. The near-fire mode turns on when the robot reaches close to a fire site. The

robot repeats this process over and over until it overcomes all obstacles and returns back to its original starting location.

3.5 Implementation Result

3.5.1 Fire Test-bed

The fire test-bed [24, 25], similar to a general indoor environment, consists of a hallway with two rooms. One room located on the side at the closed end of the hallway and another in the middle of the hallway as shown in Fig. 4. A 0.3m square sand burner powered with propane was used to produce a 20 kW diffusion flame in the larger room at the end of the hallway. This resulted in a flame height of approximately 1.0 m. The open end of the hallway is located beneath an exhaust hood to remove smoke produced by the fire inside of the test setup. Light bulbs were installed for clear view of the robot's movements and to allow the visual camera to collect all necessary data. Type K thermocouples were installed to measure gas temperatures along the ceiling and at different elevations in the two rooms and the hallway. A laser extinction system was added to measure the visibility of the smoke layer that accumulates in the hallway due to the blockage at the exit beneath the hood. The fire was ignited in the room located by the closed hallway. The hallway and rooms were heated to be over 50^{o}C before starting the test.

3.5.2 Results

Figure 22 show the visual (left) and thermal (right) images captured during the experiment. The experimental results of each numbered point in the Figure 21 will be explained as the robots progresses towards the fire and back.

① The robot is located at a start point. Because there are no obstacles at the starting area, the fire locating mode is automatically activated and the robot starts searching for an area of higher temperature. The seek-and-find algorithm guides the robot towards the entrance of the hallway.

②. When the robot senses the wall on the left of the entrance, it switches to obstacle mode and turns in order to avoid hitting this wall.

③. As it turns, it finds the midway between the walls in the hallway; it converts back to fire locating mode. However, the robot moves to left because it detects the room on the left side of the hallway and converts back to the obstacle mode again.

④. The robot becomes aware of the room but passes it because it does not pick up high enough temperatures and the incident UV light is low. Because of this, it recognizes there is no fire in there and switches to the fire locating mode. Then, the robot moves towards the entrance of the room at the end of the hallway where the highest temperature is emitted.

⑤. On the way it becomes close to the wall on its left. It changes to obstacle mode and turns slightly right to avoid the wall. Then, it changes back to fire locating mode to continue its way towards the room at the end of the hallway.

⑥. When the robot reaches the end of the hallway, it switches to obstacle mode. It senses the high temperature of the fire coming from inside the room and it turns left. When it is turning left, it becomes aware of the doorway and calculates to position itself to enter the room without hitting either sides of the doorway. However, as soon as it turns it senses the high incident UV light and changes to near-fire mode and stops. Then, the robot makes small movements left and right in order to face the fire.

⑦. The robot was able to find fire effectively. After the robot found the fire, the propane gas was turned off. It took about a minute for the fire to completely go out due to residual gas in the line. When the fire went completely, the incident UV light lowered and the robot automatically starts to move toward where the fire was because it senses a warm temperature at the former fire site.

⑧. When it senses the obstacle it changes to obstacle mode. The robot becomes aware that there is no longer a fire as it reaches the end of the room. Then it moves toward the doorway through which it entered the room.

⑨. As it exits the room, it senses the left wall at the end of the hallway and it turns right and exits the hallway to go back to the starting point of location.

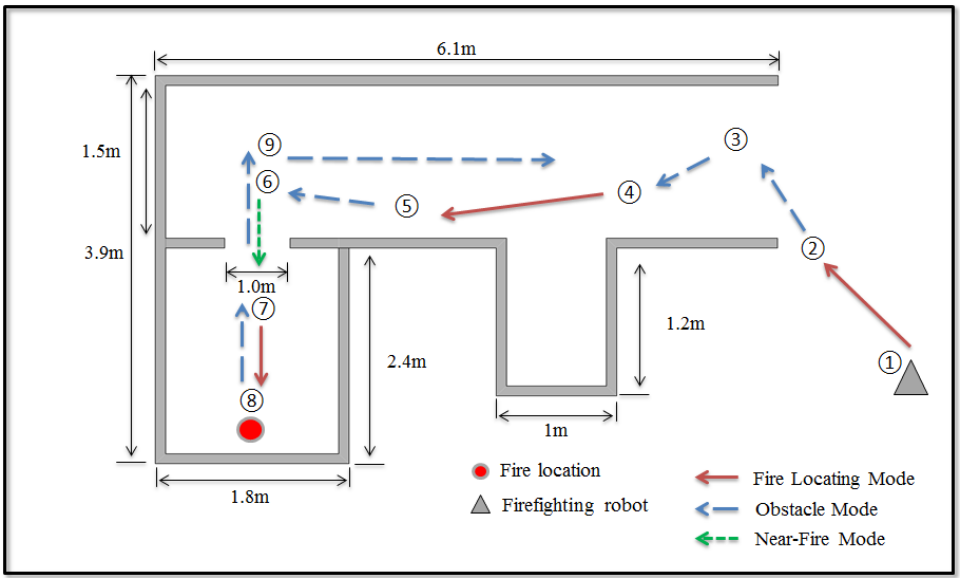
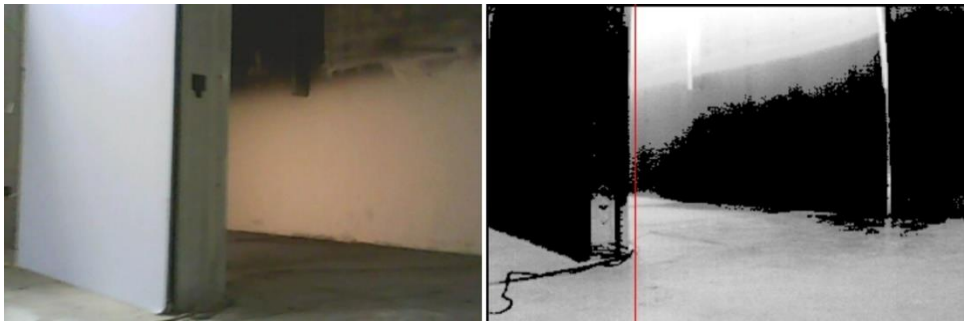
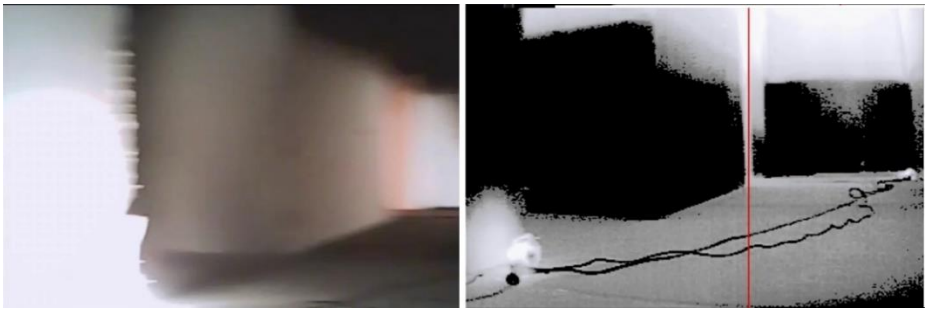


Figure 21. The fire test-bed layout and trajectory of the autonomous mobile robot platform during the experiments.

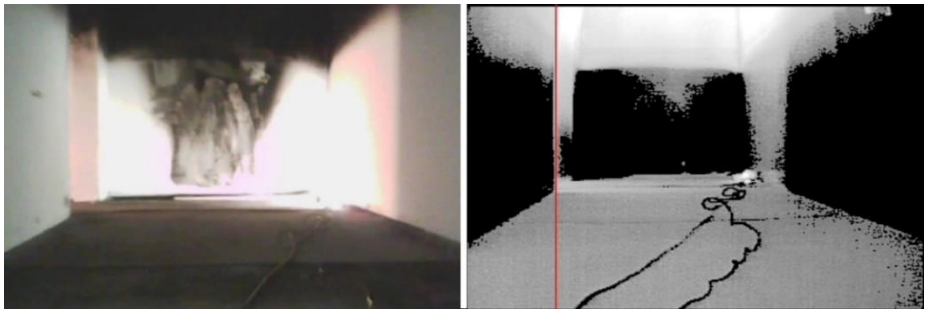


(a) The visual and thermal image at ①

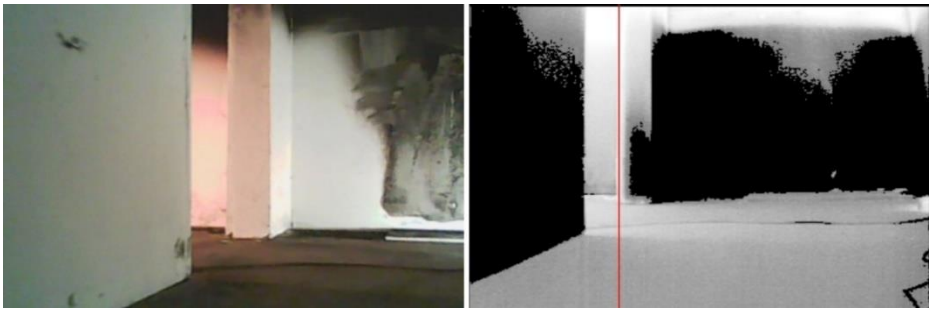
(b) The visual and thermal image at ②



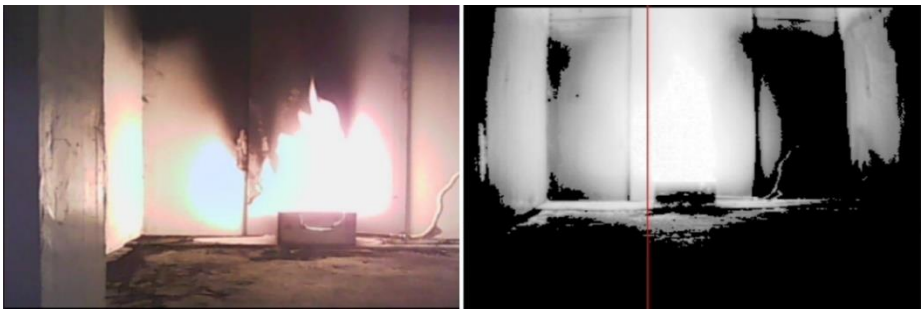
(c) The visual and thermal image at ③



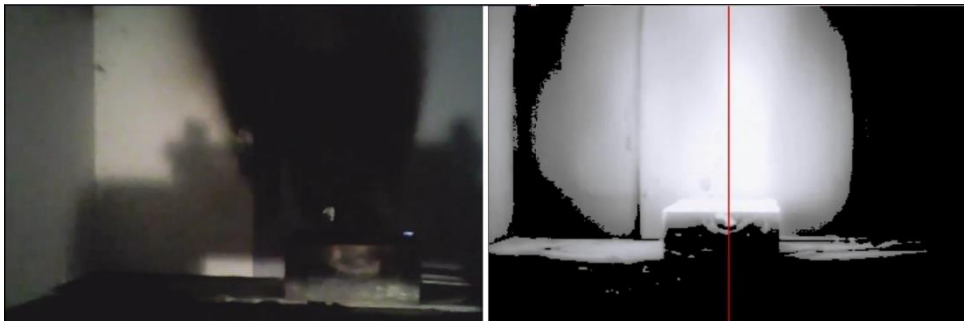
(d) The visual and thermal image at ④



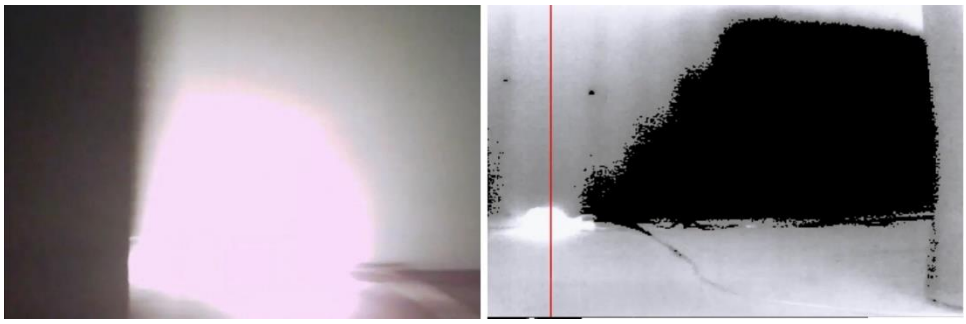
(e) The visual and thermal image at ⑤



(f) The visual and thermal image at ⑥



(g) The visual and thermal image at ⑦



(h) The visual and thermal image at ⑧

(i) The visual and thermal image at ⑨

Figure 22. Visual and thermal images at different location of the robot during the experiments.

3.6 Conclusion

This paper has presented the seek-and-find algorithm for an intelligent firefighting robot. At the instant the robot was powered on, it immediately started its move toward the direction where the fire was located. It avoided obstacles by turning accordingly. It recognized a room but passed it because there was no fire in it and ultimately found the room with the fire, stopped in front of it, and turned to face the fire. When the fire was out, it headed back to the point where it first started the search. The proposed algorithm enabled the robot to reach fire effectively by fusing sensors. The robot increased its effectiveness by switching modes that were according to each situation it encountered.

4 Thermal Image Based Feature Selection for Classification of Fire, Smoke, and Thermal Reflections

4.1 Abstract

Locating a fire inside of a structure that is not in the direct field of view of the robot has been researched for intelligent firefighting robots. By identifying fire, smoke, and their thermal reflections, firefighting robots can assess local conditions, decide a proper heading and autonomously navigate toward a fire. Long-wavelength infrared camera images were used to capture the scene due to the camera's ability to image through zero visibility smoke. This paper analyzes motion and texture features acquired from thermal images to discover the suitable features for accurate classification. Bayesian classifier is implemented to probabilistically classify multiple classes, and a multi-objective genetic algorithm optimization is performed to find the appropriate combination of the features that have the lowest errors of resubstitution and cross-validation. The distributions of multiple feature combinations that have 10% or less error were analyzed and the best solution for the classification of fire and smoke was identified.

4.2 Introduction

Intelligent firefighting robots are actively being researched to reduce firefighter injuries and deaths as well as increase their effectiveness on performing tasks. One task is locating a fire inside of a structure outside the robot field of view (FOV). Fire, smoke, and their thermal reflections can be clues to determine a heading that will ultimately lead the robot to the fire so that it can suppress it. However, research for accurately classifying these clues has been incomplete.

In conventional fire (and/or smoke) detection systems [62-66], temperature, ionization and ultraviolet light were mainly used to indicate the presence of a fire and/or smoke inside the structure, but they can have a long response time in large spaces [67] and do not provide

sufficient data for the location of fire and/or smoke. Recently using vision systems, color [68-74], dynamics [75-80] and both [67, 81-87] features have been researched to characterize fire or smoke. However, color features from RGB camera are not applicable to firefighting robots due to the fact that RGB cameras may operate in the visible to short wavelength infrared (IR) (less than 1 micron) and are not usable in smoke-filled environments where the visibility has sufficiently decreased [25, 78]. Dynamics (e.g. motion, shape changing, etc.) of the feature can be another clue to detect fire and smoke by characterizing flickering flames and smoke flow from a stationary vision system. However, the vision system onboard a robot is moving due to the dynamics of the robot itself, and this causes a large amount of noise that results in extensive computation for motion compensation. Texture features researched in Ref. [74, 77, 88, 89] were used to identify fire or smoke. The spatial characteristics of textures can be useful to recognize patterns of fire and smoke by remote sensing and are less influenced by rotation/motion [88].

Long-wavelength thermal cameras, similar to the handheld thermal infrared cameras (TIC) that are typically used to aid in firefighting tasks within smoke-filled environments [90-93], are used in this research. Due to the fact that TICs absorb infrared radiation in the long wavelength IR (7-14 microns), they are able to image surfaces even in dense smoke and zero visibility environments [25, 78]. In addition, TIC can provide proper information under local or global darkness, e.g. shadows or darkness caused by damaged lighting. The cameras will detect hot objects as well as thermal reflections off of surfaces. As a result, image processing on detected objects must be sufficiently robust to discern between desired objects and their thermal reflections.

This paper analyzes appropriate combination of features to accurately classify fire, smoke, and their thermal reflections using thermal images. Large-scale fire tests were conducted to create actual fire environments having various ranges of temperature and smoke conditions. A long wavelength IR camera was used to produce 14-bit thermal images of the fire environment. These images were used to extract motion and statistical texture features in regions of interest. Bayesian classification was performed to probabilistically identify multiple classes in real-time. To identify the best combination of features for accurate classification, the multi-objective genetic algorithm optimization was implemented using two objective functions: resubstitution and cross-validation errors.

4.3 Motion and Texture Features

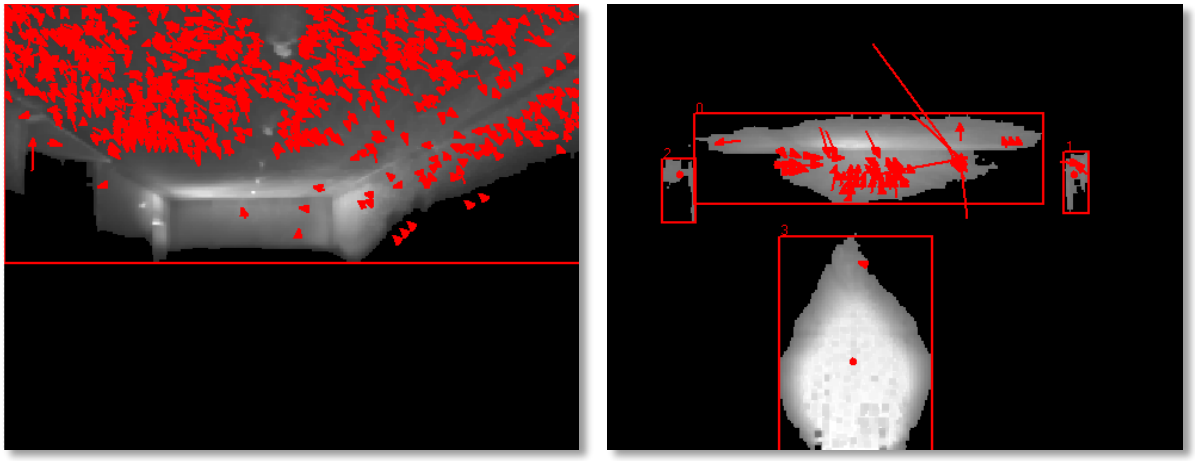
A FLIR A35 long wavelength IR camera, which is capable of imaging through zero visibility environments, was used to produce images for this study. All images were from a 320×256-pixel focal plane array, 60 Hz frame rate that produces 14-bit images with an intensity range of -16384 for -40℉ to -1 for 320℉. Twelve features from optical flow and the statistical texture features are evaluated to find the best feature combination. Optical flow shows temporal variations due to moving objects in the FOV or motion of the robot. The first and second order statistical texture features display spatial characteristics of objects in the scene.

4.3.1 Motion Features by Optical Flow

Optical flow is a useful tool to recognize motion of an object in sequential images. It consists of local and global methods. Lucas-Kanade (LK) is a local method that is relatively robust with a less dense flow field, while Horn-Schunck (HS) is a global method with a dense flow field and high sensitivity to noise [94-96]. Because the intensities in the thermal image change due to the varying fire environment, LK method that has higher robustness compared with HS was selected in this research to measure motion features of the objects. Two features of optical flow vector number (OFVN) and optical flow mean magnitude (OFVMM) were computed to quantitively characterize motions of fire, smoke and their reflections.

Figure 23 contains RGB and thermal images of dense smoke in a hallway and a wood crib fire in a room. Red arrows in the thermal images indicate the direction and magnitude of the optical flow vectors with red boxes that show smoke, fire and thermal reflections.

(a) RGB images



(b) Thermal images

Figure 23. (a) RGB images of fire scenes and (b) extracted objects from thermal images with optical flow vectors overlaid.

4.3.2 First and Second Order Statistical Texture Features

The first and second order statistical features were considered in this study for object classification. The first order statistical features estimate individual property of pixels, not characterizing any relationship between neighboring pixels, and can be computed using the intensity histogram of the candidate region of interest (ROI) in the image. Mean (MNI) and variance (VAR) of intensity were calculated by

$$\text{MNI} = \frac{1}{N_P} \sum_{i,j=1}^{N_P} I_{i,j} \tag{1}$$

$$\text{VAR} = \frac{1}{N_P} \sum_{i,j=1}^{N_P} \left(I_{i,j} - \mu\right)^2 \tag{2}$$

where I_{ij} refers to the intensity of a pixel at *i* and *j* and N_P denotes the number of pixels (NOP) of the object in the image. The second order statistical features represent spatial relationships between a pixel and its neighbors. Gray-level co-occurrence matrix (GLCM) [97] is used to account for adjacent pixel relationships in four directions (horizontal, vertical as well as left and right diagonals) by quantizing the spatial co-occurrence of neighboring pixels. A total of seven second order statistics features were used including dissimilarity (DIS), entropy (ENT), contrast (CON), inverse difference (IND), correlation (COR), uniformity (UNI) and inverse difference moment (IDM). To measure these features, a normalized co-occurrence matrix, C_{ij} is used which can be defined as,

$$C_{ij} = \frac{P_{ij}}{\sum_{i,j=1}^{N_G} P_{ij}} \tag{3}$$

where P_{ij} refers to the frequency of occurrences of two pixels of grey-level *i* and *j* within the four directions and N_G denotes the number of grey levels in the quantized image. The denominator of Eq. (3) represents the total number of *i* and *j* pairs and normalizes P_{ij} to be

estimates of the co-occurrence probabilities. After building the normalized co-occurrence matrix $C_{i,j}$, seven features of the second order statistics features were computed by

$$\text{DIS} = \sum C_{ij}|i-j|, \tag{4}$$

$$\text{ENT} = -\sum C_{ij} \log C_{ij}\,, \tag{5}$$

$$\text{CON} = \sum C_{ij}(i-j)^2\,, \tag{6}$$

$$\text{IND} = \sum \frac{C_{ij}}{1+|j-j|}, \tag{7}$$

$$\text{COR} = \sum \frac{(ij+\mu_i\mu_j-\mu_i-\mu_j)C_{ij}}{\sigma_i\sigma_j}, \tag{8}$$

$$\text{UNI} = \sum C_{ij}^{\ 2}\,, \tag{9}$$

$$\text{IDM} = \sum \frac{C_{ij}}{1+(i-j)^2} \tag{10}$$

4.4 Image Analysis and Classification

4.4.1 Adaptive Object Extraction

One of the main characteristics of fire and smoke is that they are higher in temperature than the background. With temperature related to intensity in the thermal image, fire and smoke regions appear brighter than the background. As a result, intensity is a primary factor when subtracting the foreground from the background. The intensity histogram from a thermal image changes as the surrounding environment of a fire changes. As a result, manual thresholding and

parametric modeling techniques are less efficient because they require continual updating to find the optimum parameters [98, 99]. A clustering-based image technique [100] is both auto-thresholding and nonparametric and it uses only two classes (i.e. background and foreground) resulting in fast computation during real-time implementation. It can also deal with rapidly changing environments by analyzing the intensity histogram and automatically updating an optimum threshold at each scene. Therefore, the clustering-based image auto-thresholding method is used in this research.

Assuming that the image histogram has a bimodal distribution for background and foreground, the clustering-based image auto-thresholding method can calculate an optimum threshold that minimizes the weighted sum of variances of the background and foreground for their separation. A 14-bit gray-level histogram from a FLIR A35 is normalized and considered as a probability distribution,

$$p_i = n_i / N, \quad \sum_{i=-16384}^{-1} p_i = 1 \tag{11}$$

where $n_i$is the number of pixels at level *i* and *N* is the total number of pixels. At each possible separation threshold level *k*, the occurrence probabilities of the background and foreground can be calculated as follows:

$$w_B(k) = \sum_{i=-16384}^{k} p_i \tag{12}$$

$$w_F(k) = \sum_{i=k+1}^{-1} p_i \tag{13}$$

and the mean probabilities of the background and foreground can be written by Eqs. (14) and (15).

$$\mu_B(k) = \sum_{i=-16384}^{k} i p_i / w_B(k) \tag{14}$$

$$\mu_F(k) = \sum_{i=k+1}^{-1} i p_i / w_F(k) \tag{15}$$

The between-class variance σ_{BC}^2, which is one of discriminant criterions in the discriminant analysis [101], can be calculated by Eq. (16) and the optimum threshold *k* to separate the foreground from the background can be computed by Eq. (17).

$$\sigma_{BC}^2(k) = w_B(k)w_F(k)\left[\mu_F(k) - \mu_B(k)\right]^2, \tag{16}$$

$$T_{opt} = \arg\max_{-16384 \le k \le -1}\left(\sigma_{BC}^2(k)\right) \tag{17}$$

4.4.2 Bayesian Classification

In order to simultaneously classify fire, smoke, and their thermal reflections in real-time, Bayesian classification, which is one of supervised classification in machine learning, is used in this research. Bayesian classification is Bayes' theorem-based probabilistic classification and is popular for pattern recognition applications. Although this method has lower accuracy compared with other classifiers and assumes that each feature is independent, it has fast computation, robustness to missing values and less chance of overfitting [102, 103]. In addition, this classification has the capability of probabilistic decision making over multiple classes with fast computation for real-time implementation. In this experiment, Bayesian classification is used for evaluation of each feature.

With several given features $F_1, F_2, \ldots, F_q$, (motion and statistic texture features) we can calculate the probability that one class C_j (fire, smoke and thermal reflections) corresponds to the candidate *k* by using a conditional probability, ${}^k p\left(C_j \middle| F_1 F_2 \ldots F\_q\right)$, also known as the posterior probability. By using Bayes' theorem, it can be written with prior, likelihood and evidence as shown in Eq. (18).

$${}^k p\left(C_j \middle| F_1 F_2 \ldots F_q\right) = \frac{{}^k p\left(C_j\right)^k f\left(F_1 F_2 \ldots F_q \middle| C_j\right)}{\sum_{C_j} {}^k p\left(C_j\right)^k f\left(F_1 F_2 \ldots F_q \middle| C_j\right)} \tag{18}$$

where ${}^{k}p(C_j)$ is the prior probability, meaning it represents candidate *k* probability to be C_j and can be calculated by the number of each class divided by the number of the entire class. ${}^{k}f(F_1F_2 \ldots F_q|C_j)$ is the likelihood function and the denominator of Eq. (18) is the evidence that plays as a normalizing constant by the summation of production between the prior and likelihood at each class. By applying the conditional independence assumption, the likelihood function can be rewritten by

$$f(F_1F_2 \ldots F_q|C_j) = \prod_{i=1}^{q} f(F_i|C_j) \tag{19}$$

By assuming Gaussian (or normal) distribution for features and classes, $F_i \sim N(\mu_{F_i}, \Sigma_{F_i})$ and $C_j \sim N(\mu_{C_j}, \Sigma_{C_j})$, the conditional probability density function, $f(F_i|C_j) \sim N(\mu_{F_i|C_j}, \Sigma_{F_i|C_j})$ can be described as

$$f(F_i|C_j) = \frac{1}{\sqrt{2\pi\Sigma_{F_i|C_j}}} e^{-\frac{1}{2}(F_i-\mu_{F_i|C_j})^T \Sigma_{F_i|C_j}^{-1}(F_i-\mu_{F_i|C_j})} \tag{20}$$

where

$$\mu_{F_i|C_j} = \mu_{F_i} + \Sigma_{C_jF_i}^{T} \Sigma_{C_jC_j}^{-1}(C_j - \mu_{C_i}) \tag{21}$$

$$\Sigma_{F_i|C_j} = \Sigma_{F_iF_i} - \Sigma_{C_jF_i}^{T} \Sigma_{C_jC_j}^{-1} \Sigma_{C_jF_i} \tag{22}$$

As shown in Table 4, Gaussian parameters for twelve features with respect to smoke, smoke thermal reflection, fire and fire thermal reflection were estimated by using the maximum likelihood estimation [104]. Probability density distributions for the entire features are illustrated in Figure 24. With Eq. (19), the evidence and then the posterior probability of each class were calculated. By applying the maximum priority decision rule in Eq. (23) and (24), the Bayesian classification was used to predict the class and probability of each candidate in the scene.

$$class = \arg\max_{C_j}\left(p\left(C_j \mid F_1 F_2 \ldots F_q\right)\right) \tag{23}$$

$$prob = \max\left(p\left(C_j \mid F_1 F_2 \ldots F_q\right)\right) \tag{24}$$

Table 4. Gaussian parameters.

	Smoke		**Smoke-Reflection**		**Fire**		**Fire-Reflection**	
	μ	**Σ**	**μ**	**Σ**	**μ**	**Σ**	**μ**	**Σ**
MNI	-1.2666E+04	6.9035E+02	-1.3380E+04	3.8703E+02	-5.9725E+03	1.5982E+03	-7.0840E+03	5.6409E+02
VAR	4.7822E+05	4.4880E+05	4.3927E+04	4.0967E+04	1.0489E+07	2.2865E+06	1.7044E+06	2.0444E+06
NOP	2.8866E+04	1.4148E+04	1.2536E+03	1.2030E+03	1.1018E+04	1.3589E+04	3.8337E+03	5.7472E+03
OFVN	1.0414E+03	4.0918E+03	4.8315E+01	3.5881E+02	1.8754E+02	1.6427E+03	1.1519E+02	5.1336E+02
OFVMM	8.7152E+01	5.3972E+01	1.1665E+02	8.0844E+01	1.4718E+02	2.1321E+02	1.2576E+02	6.5131E+01
DIS	7.1714E-02	1.8120E-02	2.8331E-02	1.2809E-02	1.0833E-01	8.5932E-02	5.2122E-02	4.1846E-02
ENT	4.1883E-01	9.4368E-02	1.9753E-01	8.9731E-02	3.3484E-01	1.9101E-01	1.8047E-01	1.1300E-01
CON	8.9379E-01	3.7011E-01	1.2669E-01	7.3707E-02	9.1826E-01	6.8944E-01	4.4222E-01	3.7003E-01
IND	9.8588E-01	6.4111E-03	9.9642E-01	1.6017E-03	9.6379E-01	3.6945E-02	9.8559E-01	1.5865E-02
COR	9.6558E-01	4.1639E-02	8.8492E-01	4.4602E-02	8.9344E-01	3.0867E-02	8.8608E-01	5.6502E-02
UNI	5.1059E-01	2.0174E-01	9.5052E-01	3.7404E-02	6.8359E-01	2.6599E-01	8.3693E-01	1.3943E-01
IDM	6.0061E-01	1.9192E-01	9.7429E-01	2.0729E-02	7.8188E-01	2.4229E-01	9.0796E-01	9.6689E-02

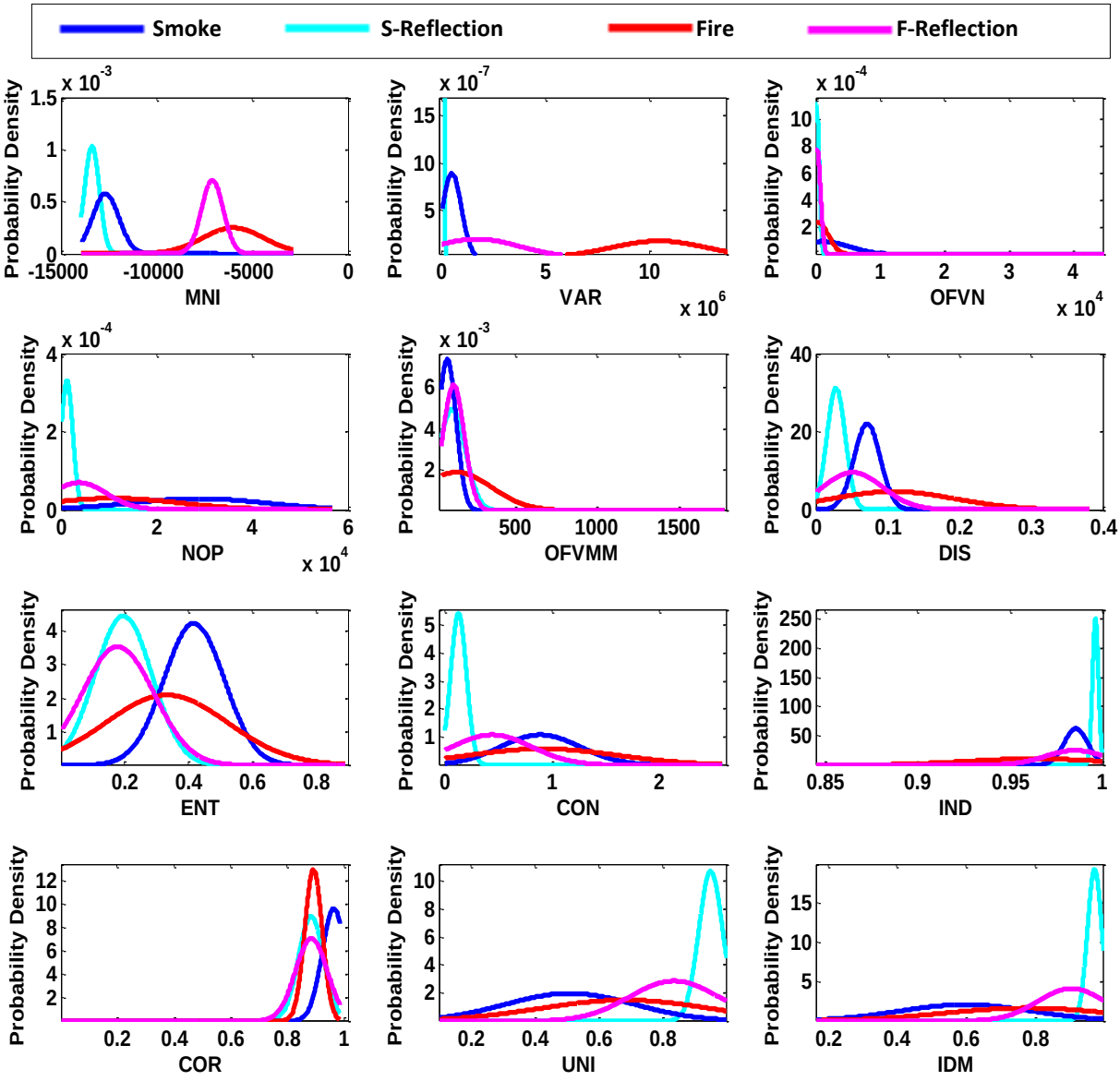


Figure 24. Probability density distributions of each features

4.5 Results

The accuracy in classifying fire objects was analyzed using data from a series of large-scale tests in the facility shown in Figure 25 using actual fires up to 75kW. Fires included latex foam, wood cribs and propane gas fires from a sand burner. These different types of fires produced a range of temperature and smoke conditions. Latex foam fires produced lower temperature conditions but dense, low visibility smoke. Conversely, propane gas fires produced higher gas temperatures and light smoke. Wood crib fires resulted in smoke and gas temperatures between that of latex foam and propane gas fires; however, these fires resulted in sparks created from the burning wood. Thermal images were collected by driving a wheeled mobile robot through the setup during a fire test. A total of 10,781 objects were collected from the experiment and because each object has thirteen corresponding data points (twelve features and a class), the total number of data points used in this paper is 140,153.

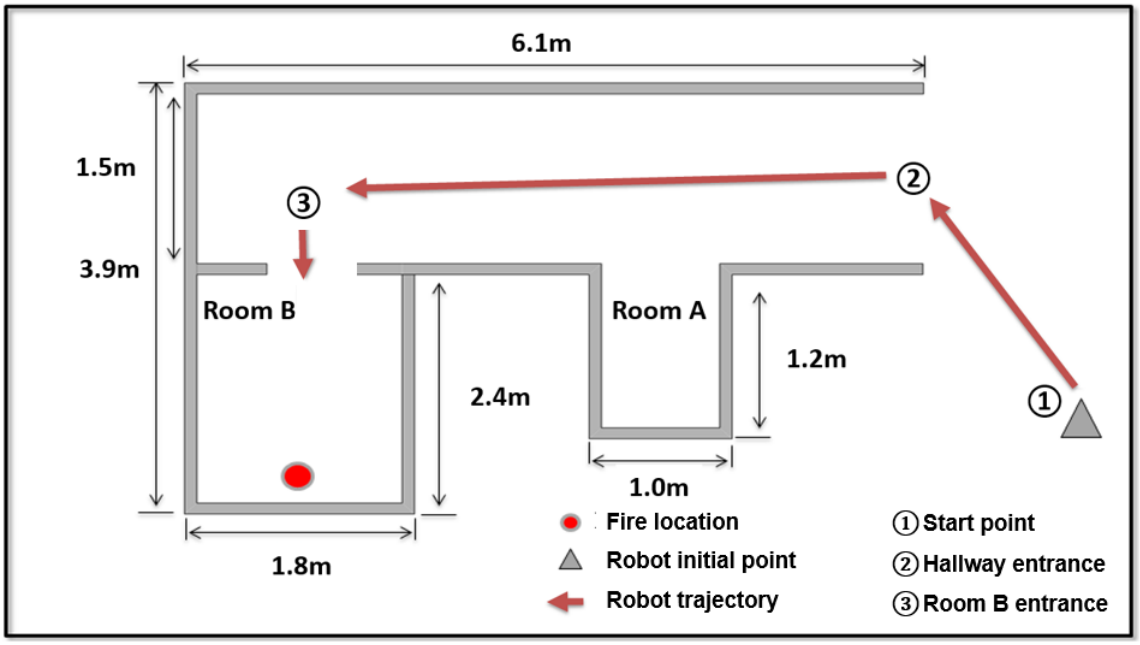


Figure 25. The test setup layout and trajectory of the robot.

Two types of error criterions (resubstitution and *k*-fold cross-validation errors [105]) were used to measure how each feature accurately performs in the classification. Resubstitution error takes the entire dataset to compare the actual classes with the predicted classes by the Bayesian classification in order to examine how well the actual and predicted classes match each other. When this criterion is used alone to enhance accuracy, the classification can be overfitted to the training dataset. Cross-validation error is advantageous to detect and prevent from overfitting. Instead of using the entire dataset, cross-validation randomly selects and splits the dataset into *k* partitions of approximately equal size ($k = 10$) to estimate a mean error by comparing between the randomly selected partition and trained results of the remaining partitions.

4.5.1 Single Feature Performance

The performance results of each feature are shown in Table 5. The first order statistical texture features produced the lowest errors while the two motion features show the highest errors compared with the other features. These results show that MNI and VAR are beneficial to distinguish fire, smoke and their thermal reflections while motion features are not. This is in part attributed to dynamic motion of the robot. ENT and COR second order statistical texture features show 42 ~ 45% error, which is higher than the other second order features.

Table 5. Performance of each feature.

	Resubstitution Error (%)	Cross-Validation Error (%)
MNI	22.9	22.9
VAR	25.9	25.9
NOP	32.4	32.5
OPVN	85.0	84.8
OFVMM	58.8	58.8
DIS	29.2	29.2
ENT	45.0	45.0
CON	29.1	29.1
IND	30.2	30.3
COR	42.9	42.9
UNI	34.9	34.8
IDM	36.0	36.0

4.5.2 Multiple Feature combination Performance

The error results in Table 5 demonstrate that a single feature cannot accurately classify fire, smoke, and their thermal reflections. Thus, possible combination of multiple features were considered and analyzed to find the best combination of the features. The total number of all possible combinations that have two or more features is

$$N_{total} = \sum_{j=2}^{m} mCj \tag{12}$$

where m refers to the total number of features (i.e. $m = 12$) and j is the number of feature combinations. Based on all possible combinations, a multi-objective genetic algorithm optimization in the global optimization toolbox of Matlab [106] was used to find the best combination of features that has the highest performance in the classification. The objective functions in the optimization, resubstitution and *k*-fold cross-validation errors [105] were used to measure how accurately different feature combinations perform in the classification.

Figure 26 contains a plot of the error associated with the most promising feature combinations. The behavioral solution set is defined as feature combinations with less than 10% error for both objective functions while the general set refers to all other possible feature combinations. The behavioral solution set contains 0.4% of all possible feature combinations.

The occurrence probability of features in the behavioral solution set is shown in Figure 27. In the behavioral solution set, the first order statistic texture features MNI and VAR always exist while OFVN and OFVMM features do not. With the experiments performed in the same facility, NOP was not included in the optimization to avoid this feature causing overfitting. Because the experiments were conducted in the same facility, this feature may cause overfitting. The second order statistic texture features ENT and COR show a higher occurrence compared with other second order texture features while IND, CON and DIS show lower occurrence.

The top features based on the probability occurrence in Figure 27 is MNI, VAR, ENT and COR. However, the combination of these four features does not result in the best solution for

classification. Table 6 contains the classification performance of the combination of features in the behavioral solution set. The best solution was determined to be a feature combination of MNI, VAR, ENT and IDM, which has the lowest of both objective functions, 9.54% and 9.38%, respectively. This combination includes all of the top features based on the probability occurrence except COR is replaced with IDM. IDM has the fifth highest probability occurrence in Figure 27.

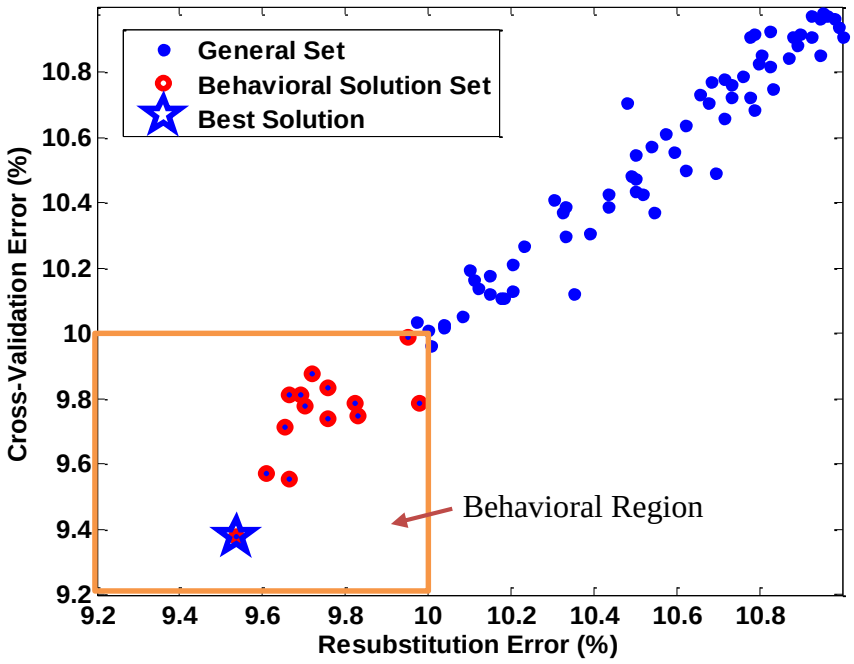


Figure 26. Multi-objective optimization result showing the general set, behavioral solution set (colored region) and the best solution.

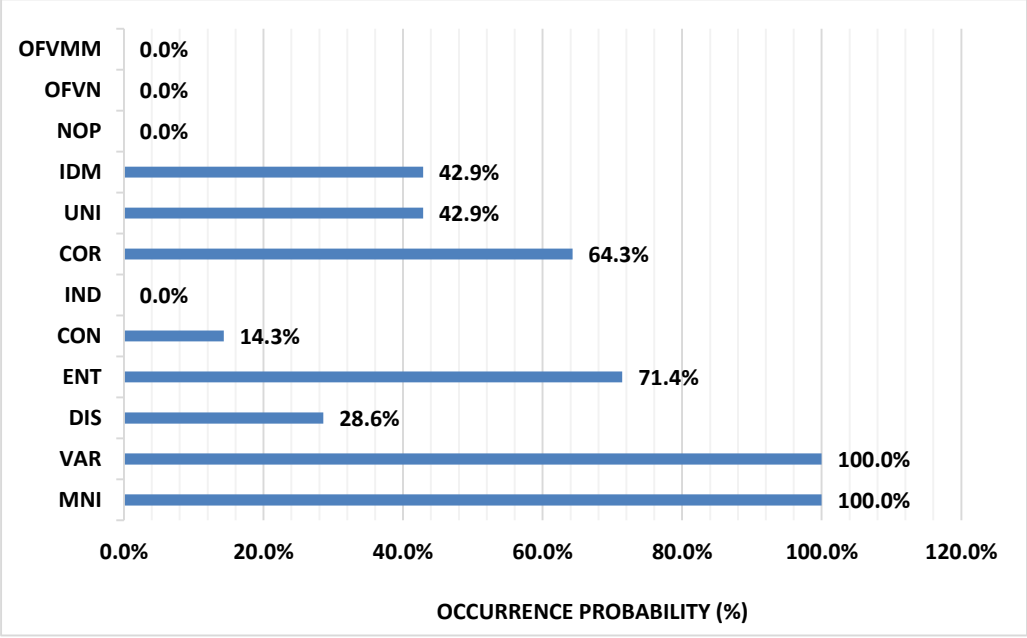


Figure 27. Occurrence analysis of the twelve feature in the behavioral region.

Table 6. Combination of features in the behavioral solution set with two error criterions.

Combination of Features	Error (%)	
	Resubstitution	Cross-Validation
MNI, VAR, UNI	9.72	9.88
MNI, VAR, IDM	9.76	9.83
MNI, VAR, ENT, CON	9.69	9.81
MNI, VAR, ENT, COR	9.61	9.57
MNI, VAR, ENT, UNI	9.83	9.75
MNI, VAR, ENT, IDM	9.54	9.38
MNI, VAR, COR, IDM	9.98	9.79
MNI, VAR, DIS, COR, UNI	9.95	9.99
MNI, VAR, ENT, COR, UNI	9.76	9.74
MNI, VAR, ENT, COR, IDM	9.67	9.55
MNI, VAR, DIS, ENT, CON, COR	9.66	9.71
MNI, VAR, DIS, ENT, COR, UNI	9.82	9.79
MNI, VAR, DIS, ENT, COR, IDM	9.67	9.81
MNI, VAR, ENT, COR, UNI, IDM	9.70	9.78

4.6 Conclusion

The appropriate combination of features was analyzed to accurately classify fire, smoke, and their thermal reflection using thermal images. Gray-scale 14-bit images from a single infrared camera were used to extract motion and texture features by applying a clustering-based, auto-thresholding technique. Bayesian classification is performed to probabilistically identify multiple classes during real-time implementation. To find the best combination of features, a multi-objective genetic algorithm optimization was implemented using resubstitution and cross-validation errors as objective functions. Large-scale fire tests with different fire sources were conducted to create a range of temperature and smoke conditions to evaluate the feature combinations.

Twelve motion and texture features were analyzed and the probability density functions of the features were computed by the maximum likelihood estimation. The combination of multiple features was determined to more accurately classify fire, smoke, and their thermal reflections compared with a single feature. In the behavioral solution set where feature combinations produce less than 10% resubstitution and cross-validation errors, MNI, VAR, ENT, and COR had 64.3% or more occurrence while DIS, CON and IND had 28.6% or less occurrence. The combination of MNI, VAR, ENT, and IDM produced the highest performance in the classification resulting in 9.54% and 9.38%, resubstitution and cross-validation errors, respectively.

5 Real-time Probabilistic Classification of Fire and Smoke using Thermal Imagery for Intelligent Firefighting Robot

5.1 Abstract

A real-time probabilistic classification method was developed for identifying fire, smoke, their thermal reflections, and other objects in infrared images. This algorithm was formulated for use on a robot that will autonomously locate fires inside of a structure where the fire is outside the robot field of view. Thermal images were used to extract features due to the fact that long wavelength infrared is capable of imaging through zero visibility environments. An accurate robot heading toward the fire requires that the robot be able to differentiate between desired characteristics, such as fire and smoke, and those that may lead the robot in the incorrect direction, such as thermal reflections and other hot objects. The probabilistic classification method in this paper provides a robust, real-time algorithm that uses thermal images to classify fire and smoke with high accuracy. The algorithm is based on four statistical texture features identified through this work to characterize and classify the candidates. Based on classification of candidates from large-scale test data, the classification performance error was measured to be 6.8% based on validation using the test dataset not included in the original training dataset. In addition, the precision, recall, F-measure, and G-measure were 93.5 – 99.9% for classifying fire and smoke using the test dataset.

5.2 Introduction

There has been interest in using robots to support firefighters in order to reduce firefighter deaths and increase their effectiveness on the job. Various sizes and shapes of firefighting robots have been developed for outdoor and indoor purposes [12, 15, 18, 20, 107, 108]; however, most of them are remotely controlled by operators. Recently, a shipboard autonomous firefighting robot (SAFFiR) program, prototype is illustrated in Figure 28, is being developed to locate and

suppress fires inside ships and structures. Through the SAFFiR program, further advancements of artificial intelligent algorithms / perception systems [22-26] and unmanned fire suppression systems [27] have been developed to enhance autonomous firefighting robots.

One such task is locating a fire inside of a structure that is not in the direct field of view (FOV) of the robot. This involves using features (e.g., presence of smoke, flames) in the FOV of the robot to determine a heading that will ultimately lead the robot to the fire so that it can suppress it. Many conventional detection systems will effectively indicate the presence of a fire inside the structure [63, 65, 66], but do not provide sufficient data to provide the robot with a heading. Extracting fire and smoke features from an environment has been explored for advanced detection systems [82, 84, 86, 87, 109]. However, the existing fire and smoke detection methods [69, 78, 79, 84, 87, 109, 110] have practical limitations that make them not directly applicable for use on a dynamic robot.

Vision systems using RGB cameras have been widely used for fire (or smoke) detection due to their low weight, low power consumption and large FOV that includes color, shape, motion and texture information. The majority of these applications use fixed location surveillance cameras for smoke and fire detection. RGB image databases online are used to support algorithm development [111-113]. Camera-based methods reported in Refs. [69, 73, 80, 86, 109, 110] are not applicable to firefighting robots due to the high false positive rate from colors or reflection illuminations similar to that with fire [78]. Due to the fact that RGB cameras may operate in the visible to short wavelength infrared (IR) (less than 1 micron), they are not usable in smoke-filled environments where the visibility has sufficiently decreased [25, 78]. In addition, because the performance of the cameras depends on light, these methods cannot provide proper information under local or global darkness, e.g. shadows or darkness caused by damaged lighting.

Handheld thermal infrared cameras (TIC) are typically used to aid in firefighting tasks within smoke-filled environments [90-93]. Due to the fact that TICs absorb infrared radiation in the long wavelength IR (7-14 microns), they are able to image surfaces even in dense smoke and zero visibility environments [25, 78]. TICs produce grey scale images; therefore, online RGB image databases cannot be used to support development of feature extraction and classification algorithms [111-113]. Unlike RGB cameras, limited studies have investigated the use of thermal images to extract fire features such as smoke or flames to assist in locating fires.

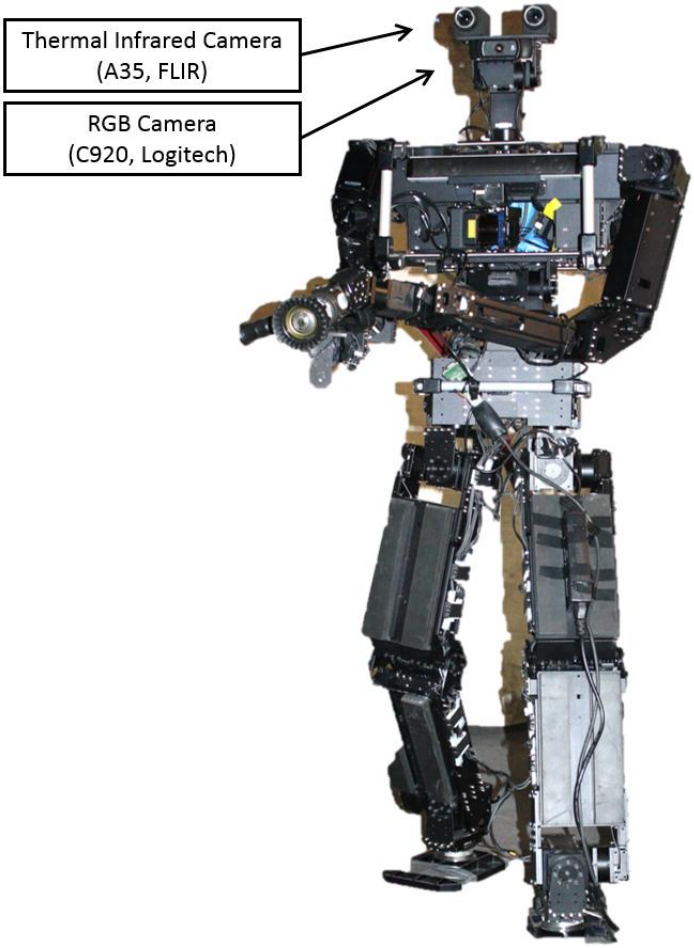


Figure 28. A prototype of the shipboard autonomous firefighting robot (SAFFiR) with two thermal infrared cameras and a RGB camera on its head.

The vision system onboard a robot will contain cameras that are moving due to the dynamics of the robot itself. As a result, there are limitations in using methods that incorporate relative motion features from a stationary camera [82, 84, 86, 87, 109]. In a previous study conducted on locating fires using thermal images [24], there were problems distinguishing between fire reflections and an actual fire as well as smoke reflections and smoke when they appeared together in the FOV. Due to this, erroneous robot headings were provided which led the robot in the wrong direction during navigation. Existing research on binary classification, which classifies fire from a non-fire object [69, 78, 86] or smoke and non-smoke object [79, 87,

88], has not been able to distinguish fire and smoke from their reflections or classify fire, smoke, and reflections simultaneously. This information is needed to provide the robot with a more accurate heading for navigation toward a fire outside the FOV.

This paper presents a probabilistic classification method for fire, smoke, their thermal reflections, and other hot objects (i.e., not fire, smoke, or reflections). The method uses images from a single thermal infrared camera mounted on a robot and acquires textural features to recognize patterns while minimizing the influence of dynamic robot motion. Bayes' theorem was used to compute the posterior probabilities of the fire, smoke, thermal reflections, and other object classes, which can be used to assist in deciding the optimal heading for navigating toward the fire. For reliability, the method was trained using large-scale fire test data having a range of temperature and smoke conditions. The real-time algorithm was then tested using a separate series of tests to validate its ability to accurately classify the fire environment.

5.3 System Architecture

Fire, smoke, their thermal reflections and other objects (i.e., not fire, smoke, or reflections) are probabilistically classified to provide SAFFiR with information to determine a proper heading and autonomously navigate toward a fire that is outside the robot FOV. As illustrated in Figure 29, the proposed method consists of extracting relevant candidates in the scene and probabilistically classifying the candidates based on statistical texture features. Images for analysis were taken using a FLIR A35 long wavelength infrared (7-14 micron) camera with a 320×256-pixel focal plane array, 60 Hz frame rate that produces 14 bit images with intensity ranges between -16384 for -40℃ and -1 for 160℃. This particular camera was used since it is lightweight and can image in clear as well as zero visibility fire smoke conditions. Candidates from the images that have the highest likelihood to be fire, smoke, or reflections are extracted from the scene by adaptive background subtraction and morphological filtering. However, it is difficult to distinguish fire and smoke from their reflections through these methods alone, and this may cause erroneous headings leading the robot in the wrong direction during navigation. To overcome this, Bayesian classification was performed to probabilistically distinguish fire and smoke from their reflections by analyzing multiple texture features extracted from the

candidates. Extracted candidates that are determined to not be fire, smoke, or thermal reflections are labeled as other objects. Adaptive histogram equalization [114] and colored ellipse overlay were applied to produce final images.

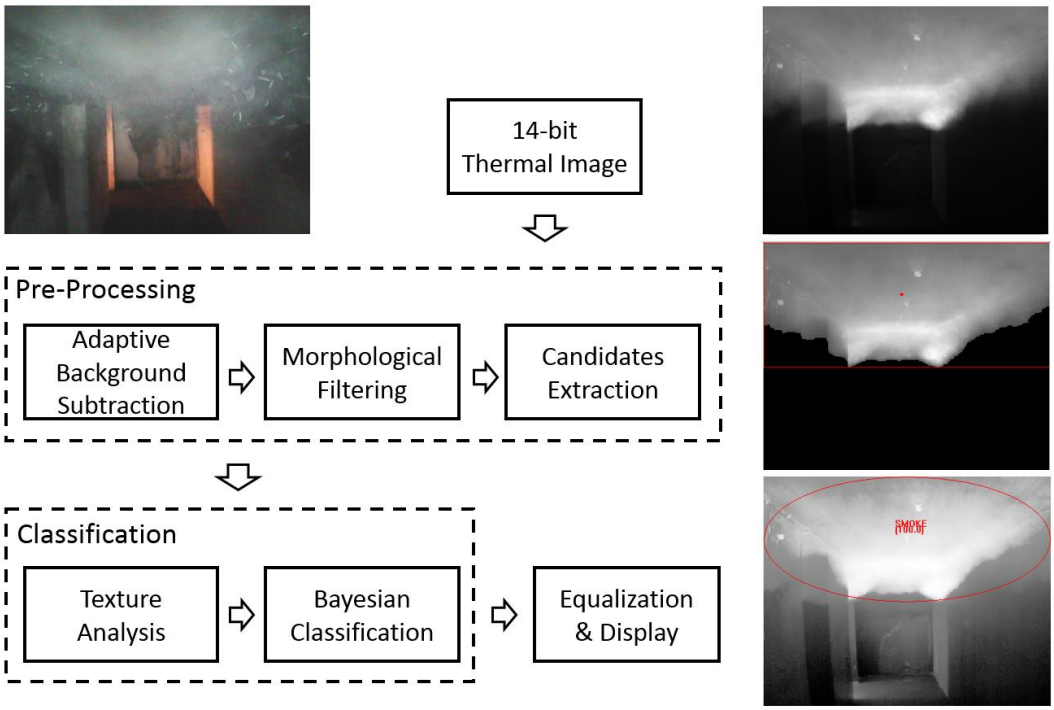


Figure 29. System architecture with thermal images at each stage.

5.4 Pre-Processing

5.4.1 Adaptive Background Subtraction

Existing research [82, 84, 86, 87, 109, 115] uses color and/or motion-based background subtraction from a stationary camera to detect the foreground, which contains the candidates of interest. However, these approaches are not compatible with this research because color information is not available in thermal images and retrieving relative motion from robot in motion creates a large amount of noise that results in extensive computation for motion compensation. One of the main characteristics of fire and smoke is that they are higher in temperature than the background. With temperature related to intensity in the thermal image, fire

and smoke regions appear brighter than the background. As a result, intensity is a primary factor when subtracting the foreground from the background.

The intensity histogram from a thermal image changes as the surrounding environment of a fire changes. As a result, manual thresholding and parametric modeling techniques are less efficient because they require continual updating to find the optimum parameters [98, 99]. A clustering-based image technique [100] is both auto-thresholding and nonparametric and it uses only two classes (i.e. background and foreground) resulting in fast computation during real-time implementation. It can also deal with rapidly changing environments by analyzing the intensity histogram and automatically updating an optimum threshold at each scene. Therefore, the clustering-based image auto-thresholding method is used in this research.

Assuming that the image histogram has a bimodal distribution for background and foreground, the clustering-based image auto-thresholding method can calculate an optimum threshold that minimizes the weighted sum of variances of the background and foreground for their separation. A 14-bit gray-level histogram from a FLIR A35 is normalized and considered as a probability distribution as described in Eq. (1)

$$p_i = n_i / N, \quad \sum_{i=-16384}^{-1} p_i = 1 \tag{1}$$

where $n_i$is the number of pixels at level *i* and *N* is the total number of pixels. At each possible separation threshold level *k*, the occurrence probabilities of the background and foreground can be calculated as follows:

$$w_B(k) = \sum_{i=-16384}^{k} p_i \tag{2}$$

$$w_F(k) = \sum_{i=k+1}^{-1} p_i \tag{3}$$

and the mean probabilities of the background and foreground can be written by Eqs. (4) and (5).

$$\mu_B(k) = \sum_{i=-16384}^{k} ip_i / w_B(k) \tag{4}$$

$$\mu_F(k) = \sum_{i=k+1}^{-1} ip_i / w_F(k) \tag{5}$$

The between-class variance σ_{BC}^2, which is one of discriminant criterions in the discriminant analysis [101], can be calculated by Eq. (6) and the optimum threshold *k* to separate the foreground from the background can be computed by Eq. (7).

$$\sigma_{BC}^2[k] = w_B(k) w_F(k) [\mu_F(k) - \mu_B(k)]^2, \tag{6}$$

$$T_{opt} = \arg\max_{-16384 \le k \le -1} (\sigma_{BC}^2(k)) \tag{7}$$

5.4.2 Morphological Filtering and Extracting Candidates

The optimum threshold computed by the clustering-based image auto-thresholding method was used to extract fire, smoke and reflection candidates from the background, creating a binary image with 0 being the background and 1 being all candidates (foreground). Binary images were filtered to remove small regions of interest (ROI) in the image and holes inside candidate ROI were removed through morphological filtering techniques. By convoluting the original 14-bit image with the filtered-binary image, an image was obtained that includes the original 14-bit intensities in candidate ROI as well as zeroes in the rest of the image. Small ROI candidates with less than a specified number of pixels were found to not provide reliable texture features and were ignored.

5.5 Fire, Smoke, Thermal Reflection and Other Objects Classification

5.5.1 Statistical Texture Features

Pre-processing provided candidates regions that have a high likelihood of being fire, smoke, or their thermal reflections. In order to characterize these candidates, the statistical texture features are analyzed due to the fact that texture features are less influenced by rotation/motion [88]. In addition, textures of a 14-bit gray level have high-resolution spatial distribution, which is useful to recognize patterns of fire, smoke, and their thermal reflections for real time classification.

A description of cluttered thermal images is determined using the first order statistical textures such as mean and variance, which are computed using the intensity histogram of the candidate ROI in the image. For second order statistical textures, the gray-level co-occurrence matrix (GLCM) [97] is used and accounts for adjacent pixel relationships with four directions (horizontal, vertical, left and right diagonals) by quantizing the spatial co-occurrence of neighboring pixels. The third and higher order statistic features investigated in Ref. [116] are not considered in this research due to computational expense of applying these.

The first order statistical features estimate the individual property of pixels and do not consider any relationship between neighboring pixels, whereas the second order statistical features represent spatial relationships between a pixel and its neighbors. Multi-objective optimization [106] was performed to determine the appropriate first and second order texture features [97, 117, 118] to be used in the classification of fire, smoke, and reflections. Through this optimization, it was found that mean and variance first order texture features as well as entropy (ENT) and inverse difference moment (IDM) second order texture features should be used for classification of smoke, fire and reflections. Mean is the average intensity showing brightness and darkness of the candidates while variance represents the deviation of intensity from the mean in regions of candidates. ENT characterizes statistical randomness of texture and IDM measures the closeness of distribution in GLCM [97] [119]. While the mean and variance features can be calculated by grey-level values in the candidate ROI, ENT and IDM were computed by the normalized co-occurrence matrix,

$$C_{ij} = \frac{P_{ij}}{\sum_{i,j=1}^{N_G} P_{ij}} \tag{8}$$

where P_{ij} refers to the frequency of occurrences of two pixels of grey-level i and j within the four directions; N_G denotes the number of grey levels in the quantized image. The denominator of Eq. (8) represents the total number of i and j pairs and normalizes P_{ij} to be estimates of the co-occurrence probabilities. After building the normalized co-occurrence matrix $C_{i,j}$, ENT and IDM were calculated as follows :

$$\text{ENT} = \sum C_{ij} \log C_{ij}\,, \tag{9}$$

$$\text{IDM} = \sum \frac{C_{ij}}{1+(i-j)^2} \tag{10}$$

ENT outputs a higher value for a random distribution of $C_{i,j}$. IDM generates higher values for an image containing large homogeneous patches by weighting $1/1 + (i-j)^2$ on $C_{i,j}$, and this offers lowest weights to $C_{i,j}$ when located away from the main diagonal.

These statistical texture features are useful to recognize patterns of fire, smoke, their reflections, and other objects. Flames generate a local maximum intensity in the 14-bit histogram (intensity value of -1) at the core of the flame as well as a range of high intensities associated with the variable flame temperature outside the flame core. As a result, flames create higher mean intensity and variance compared with other classes. Smoke is lower temperature than fire and no local maximum; therefore, the intensity and variance levels of smoke are lower than fire. However, the ENT of smoke is higher than fire due to more random distribution in texture, a distinctive feature of smoke. The amount of reflected energy depends on surface roughness of material with incidence angle [117]. As a result, thermal reflections of fire and smoke have lower intensity and variance as well as higher IDM compared with fire and smoke. These four statistical texture features were used to classify fire, smoke and their reflections by statistically characterizing the distinctive features of the candidates.

5.6 Bayesian Classification

Classification methods in supervised machine learning were considered to train and classify candidates. The *k*-nearest neighbors (*k*NN) uses *k* number of nearest patterns to a target pattern to classify based on a majority vote [102, 120]. While *k*NNs are insensitive to outliers, they require a large amount of memory and are computationally expensive when large amounts of data exist [102]. Decision tree (DT) methods [121] are tree models that predict the value of targets using branches and nodes during the decision process. DTs have low computation burden but in multi-label classification with of a large data set, they may produce a complicated tree structure and may result in the overfitting problem [102, 122]. Naïve Bayesian is Bayes' theorem-based probabilistic classification. This method has lower accuracy compared with other methods and assumes that each feature is independent; however, it is computationally fast, robust in analysis of missing values and not prone to overfitting [102, 103]. Neural networks (NN) work much better when dealing with multi-dimensions and continuous features [103]. Unlike naïve Bayesian, NNs do not have strong assumptions about probability distributions. Although NNs have fast classification computation and flexibility to system complexity, they are inefficient in dealing with missing or irrelevant values and cannot overcome the overfitting problem [103]. Support vector machines (SVMs) are discriminative classification methods that find a separating hyperplane with the greatest possible margin. SVMs provide fast computation and the highest accuracy among classification methods [103]. However, because SVMs generate binary results, it cannot be used to perform multi-label classification. The Bayesian classification was chosen for this research due to its probabilistic decision making over multiple classes, lower chance of overfitting, and fast computation for real-time implementation.

With several given features $F_1, F_2, \ldots, F_q$, the probability that class C_j corresponds to a candidate *k* can be calculated by using a conditional probability, ${}^k p\left(C_j \middle| F_1 F_2 \ldots F\_q\right)$, also known as the posterior probability. By using Bayes' theorem, the posterior probability is calculated by

$$ {}^k p\left(C_j \middle| F_1 F_2 \ldots F_q\right) = \frac{{}^k p\left(C_j\right) {}^k f\left(F_1 F_2 \ldots F_q \middle| C_j\right)}{\sum_{C_j} {}^k p\left(C_j\right) {}^k f\left(F_1 F_2 \ldots F_q \middle| C_j\right)} \qquad (11) $$

where ${}^{k}p(C_j)$ is the prior probability, which represents the probability that candidate *k* is in class C_j. For a given dataset, the prior probability can be calculated by the number of candidates in each class divided by the total number of candidates in the dataset. ${}^{k}f(F_1F_2 \ldots F_q|C_j)$ is the likelihood function and the denominator of Eq. (11) is the evidence. By applying the conditional independence assumption, the likelihood function can be rewritten by

$$f(F_1F_2 \ldots F_q|C_j) = \prod_{i=1}^{q} f(F_i|C_j) \tag{12}$$

By assuming Gaussian (or normal) distribution for features and classes, $F_i \sim N(\mu_{F_i}, \Sigma_{F_i})$ and $C_j \sim N(\mu_{C_j}, \Sigma_{C_j})$, the conditional probability density function, $f(F_i|C_j) \sim N(\mu_{F_i|C_j}, \Sigma_{F_i|C_j})$ can be described as

$$f(F_i|C_j) = \frac{1}{\sqrt{2\pi\Sigma_{F_i|C_j}}} e^{-\frac{1}{2}\left(F_i-\mu_{F_i|C_j}\right)^T \Sigma_{F_i|C_j}^{-1}\left(F_i-\mu_{F_i|C_j}\right)} \tag{13}$$

Where

$$\mu_{F_i|C_j} = \mu_{F_i} + \Sigma_{C_jF_i}^{T}\Sigma_{C_jC_j}^{-1}\left(C_j - \mu_{C_i}\right) \tag{14}$$

$$\Sigma_{F_i|C_j} = \Sigma_{F_iF_i} - \Sigma_{C_jF_i}^{T}\Sigma_{C_jC_j}^{-1}\Sigma_{C_jF_i} \tag{15}$$

Supervised training was used to determine the class of the data. The Gaussian parameters for four features with respect to each class were calculated by using the maximum likelihood estimation [104] and are shown in Table 7. The posterior probability of each class is calculated using Eq. (11) with the prior probability of each class from the dataset and the likelihood determined using Eqs. (12) - (15). The Bayesian classification can predict the class and probability of each candidate in the scene using the maximum priority decision rule through the following relations:

$$class = \arg\max_{C_j}\left(p\left(C_j \mid F_1 F_2 \ldots F_q\right)\right) \tag{16}$$

$$prob = \max\left(p\left(C_j \mid F_1 F_2 \ldots F_q\right)\right) \tag{17}$$

Table 7. Gaussian parameters

Class	Mean Intensity		Variance		Entropy		IDM	
	μ	Σ	μ	Σ	μ	Σ	μ	Σ
Smoke	-1.2666E+04	6.9035E+02	4.7822E+05	4.4880E+05	4.1883E-01	9.4368E-02	6.0061E-01	1.9192E-01
S-Reflection	-1.3380E+04	3.8703E+02	4.3927E+04	4.0967E+04	1.9753E-01	8.9731E-02	9.7429E-01	2.0729E-02
Fire	-5.9725E+03	1.5982E+03	1.0489E+07	2.2865E+06	3.3484E-01	1.9101E-01	7.8188E-01	2.4229E-01
F-Reflection	-7.0840E+03	5.6409E+02	1.7044E+06	2.0444E+06	1.8047E-01	1.1300E-01	9.0796E-01	9.6689E-02

The candidates in the image are geometrically highlighted by colored ellipses. The class and probability of each candidate are displayed at the center of the ellipses. The red represents smoke and fire while the orange indicates a thermal reflection. In addition, there is a bias problem in the thermal image histogram. To solve this, adaptive histogram equalization [114] was implemented for redistributing pixel values and improving the image display.

5.7 Experiments

The proposed method was trained and evaluated in a large-scale test facility with actual fires up to 75kW. Fires included latex foam, wood cribs and propane gas fires from a sand burner. These different types of fires were used to produce a range of temperature and smoke conditions. A total of 10 videos having 11,964 frames with 15,381 candidates were collected from the experiments and divided into two different datasets, a training dataset and a test dataset. The training dataset included 8,425 frames with 10,781 candidates, which was about two thirds of the total data. The test dataset used to evaluate the algorithm consisted of 3,529 frames with 4,600 candidates, which was the remaining one third of the data. Because each candidate has five corresponding data points (four features and a class), the total number of data points used in this paper is 76,905. The training dataset was only used to estimate parameters of the classification model while the test dataset was used to validate the performance of the classification model.

5.7.1 Experimental Setup

The experimental setup consisted of a hallway with two adjacent rooms to create a general indoor environment as shown in Figure 30. The setup was instrumented with thermocouples located 12 mm below the ceiling at nine different locations to measure gas temperatures throughout the setup. Two laser extinction systems (using HeNe lasers operating at 632 nm) were used to monitor the visibility through the fire smoke at 0.9 m below the ceiling. During the experiments, the visibility in the setup dropped below 1.0 m. The temperature at the ceiling in the hallway remained below 150°C and below 70 °C at the location of the robot head where the FLIR A35 cameras were mounted. The fire was ignited inside Room B located by the closed end

of the hallway. The open end of the hallway is located beneath an exhaust hood, which removed smoke from the laboratory.

5.7.2 Experimental Scenario

Three different fire sources were used in these experiments : 150 mm × 150 mm × 150 mm latex foam blocks, 300 mm × 300 mm × 200 mm wood cribs, and propane gas fires from a 300 mm x 300 mm sand burner. These different sources of fire created different levels of fire conditions. Latex foam fires produced lower temperature conditions but dense, low visibility smoke. Conversely, propane gas fires produced higher gas temperatures and light smoke. Wood crib fires resulted in smoke and gas temperatures between that of latex foam and propane gas fires; however, these fires resulted in sparks created from the burning wood.

A remote controlled mobile wheeled robot was used collect the perception data in the testing. The thermal infrared cameras were installed at an elevation of 1.75 m, which is similar to the final height planned on the SAFFiR robot. The mobile robot was placed outside of the hallway in the main lab space. The robot started moving when it observed conditions perceived as being consistent with a fire, such as the presence of smoke. Details on the robot initial location, fire location, and robot trajectory toward the fire are illustrated in Figure 30.

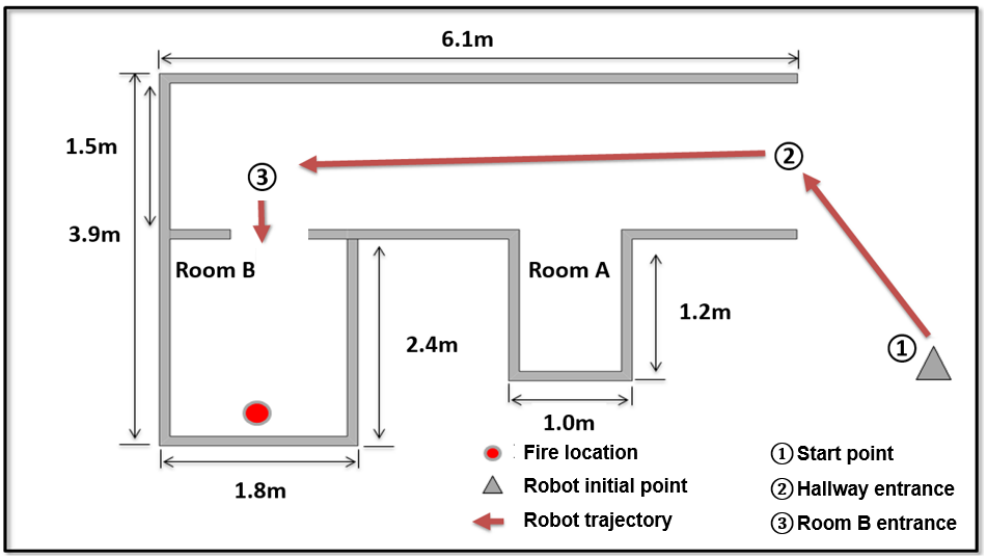


Figure 30. The test setup layout and trajectory of the robot.

5.8 Results

This research consists of pre-processing and classification for the purpose of firefighting robots to successfully find a fire outside the FOV. During pre-processing, candidates that have a high likelihood of being fire, smoke, or thermal reflection were extracted. During classification, statistical texture features were analyzed and the posterior probability of the five classes was estimated through Bayesian classification. The entire processing was performed using an i7, 2.5 GHz PC with a 1 GB graphic card and 8 GB of RAM. The proposed method processes each thermal image in approximately 20 ms for real-time implementation. The algorithm was developed and tested using data from three fire sources, which generate different realistic fires, temperatures, and smoke conditions.

5.8.1 Result Display

Figure 31 shows visual and thermal images from the different fire sources with the robot at different locations in the test setup. Each column corresponds to a series of images from the robot at a particular location for a specific test (e.g., the first row is from a propane fire with the robot at the end of the hallway). Figure 31(a1) contains an RGB image of the robot view at the hallway entrance below the exhaust hood with a propane fire in the room. As seen in the RGB image, further information regarding the hallway is limited due to shadowing of the light. Figure 31(a2) contains an RGB image of a smoke layer in the upper portion of the hallway due to a latex foam fire inside the room. Similar to a real fire situation, the smoke is thick making it difficult for the upper portion of the hallway to be apparent using the RGB cameras. In Figure 31(a3), the RGB image depicts a wood crib fire with sparks. Due to soot and relative difference in brightness, the background is shown darker and thus limiting information on the background around the fire.

Thermal images are seen in the second row to display information that visual RGB images cannot provide in fire environments. Unlike the hallway entrance in Figure 4(a1) that is obscured due to shadowing, the presence of smoke and thermal reflections on the ventilation

hood can be clearly seen in Figure 4(b1). In Figure 4(b2) and 4(b3), thermal images provide the images of smoke and fire as well as background information that is otherwise not visible through visual imaging.

Results of candidate extraction through pre-processing of the thermal images are provided in the third row and highlighted in the image by boxing the candidates with identification numbers. On the fourth row, class labels and posterior probabilities of each candidate are marked at the center of a candidate ROI as a result of classification. Smoke and fire are marked in red ellipses and their reflections are marked in orange. Lastly, the final thermal images in Figures 4(d1), 4(d2) and 4(d3) are more refined and clearer than thermal images in Figures 4(b1), 4(b2) and 4(b3) due to adaptive histogram equalization.

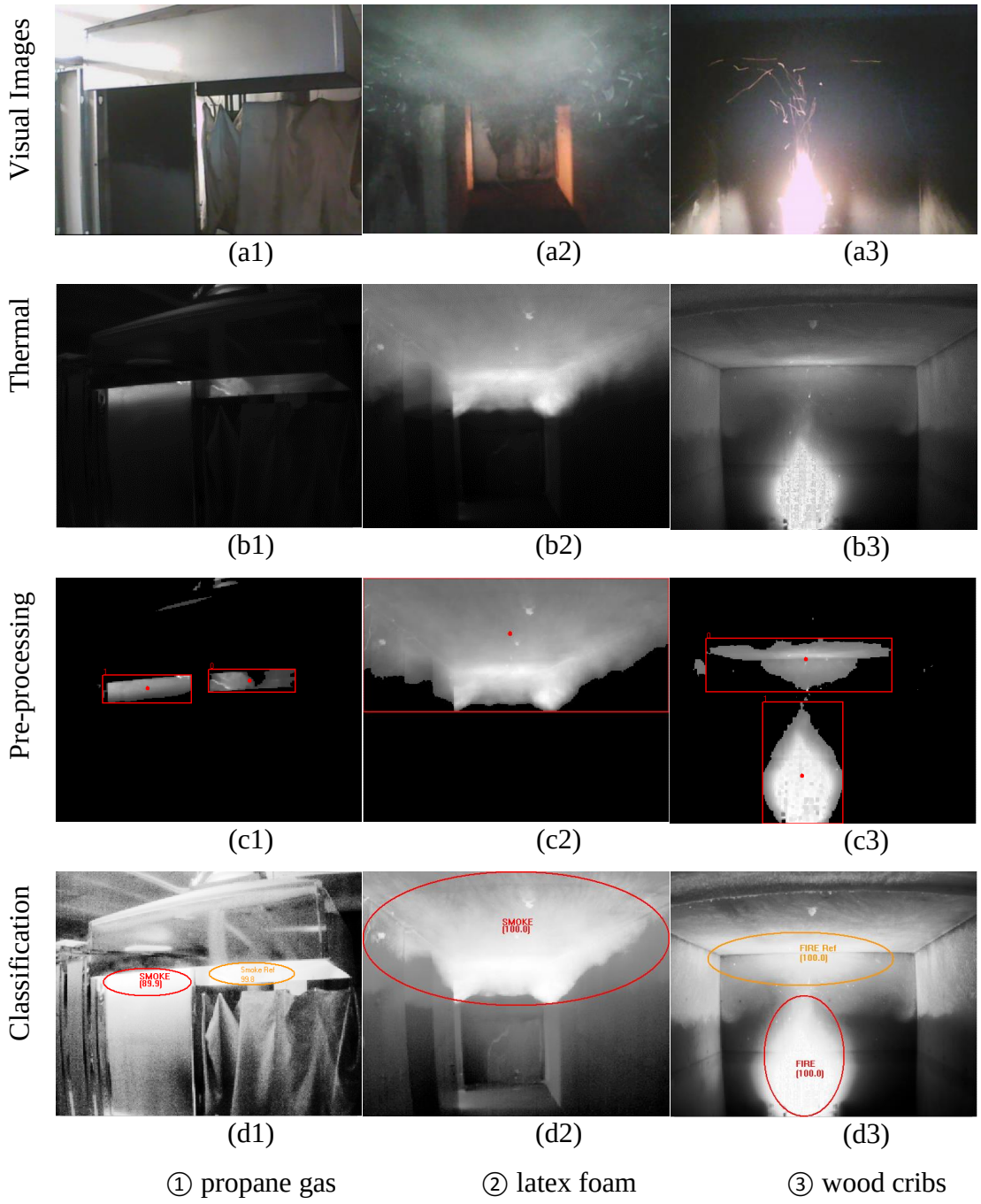


Figure 31. Visual and thermal images of three different fire sources (propane gas, latex foam and wood crib) at three different locations (entrance of the hallway ①, middle of the hallway ② and front of fire ③).

5.8.2 Analysis

In order to measure how accurately this proposed method will perform in practice, two types of error criterions were applied in this paper to analyze classification results; resubstitution and *k*-fold cross-validation errors [105]. Resubstitution error uses the entire dataset to compare the actual classes with the predicted classes by the proposed method to examine how well they match each other. When this criterion is used alone to enhance accuracy, the classification can be overfitted to the training dataset. On the other hand, cross-validation error is useful to detect and prevent from overfitting. Instead of using the entire dataset, cross-validation randomly selects and splits the dataset into *k* partitions of approximately equal size (10 was used in this research) to estimate a mean error by comparing between the randomly selected partition and trained results of the remaining partitions. Table 8 shows the resubstitution and cross-validation errors estimated from the training dataset of seven various fire experiment videos. Videos 1 to 4 are latex foam fires, Video 5 is a wood crib fire and Videos 6 and 7 are propane gas fires. Although there is less than 10% error in general, Video 4 had higher error rates due to the misclassification that occurred between smoke and other objects (i.e., not fire, smoke, or thermal reflections). This is believed to be due to the fire in this test being small, resulting in small amounts of smoke and low temperatures. Video 6 has resubstitution and cross-validation errors of 21.3% and 8.69%, respectively. These errors occurred mostly during smoke and smoke reflection classification under small amount of smoke conditions. The entire performance of the training dataset of the seven videos, labeled Total in Table 8, are evaluated at 9.54% and 9.38% for the resubstitution and cross-validation errors, respectively.

Table 8. Training dataset results

Videos	Frames	Candidates	Sources	Resubstitution Error (%)	Cross-Validation Error (%)
Video #1	1,771	2,064	Latex foam	4.26	1.89
Video #2	1,730	1,924	Latex foam	1.92	1.61
Video #3	1,684	1,853	Latex foam	1.57	1.03
Video #4	695	934	Small Latex	51.5	0.96
Video #5	1,120	1,953	Wood crib	7.42	5.58
Video #6	700	1,024	Propane gas 25kW	21.3	8.69
Video #7	725	1,029	Propane gas 75kW	2.90	5.83
Total	**8,425**	**10,781**		**9.54**	**9.38**

The performance of the proposed method was also validated using the test dataset, which was not part of the training dataset. Table 9 shows the test dataset results. Videos 8, 9 and 10 are fires of latex foam, wood crib and propane gas, respectively. The resubstitution errors are 3.05%, 4.09% and 12.04% while the cross-validation errors are 3.88%, 6.08% and 5.48%, respectively. The entire performance of the test datasets is assessed at 6.87% of the resubstitution error and 6.78% of the cross-validation error.

Table 9. Test dataset results

Video	Frame	Candidates	Source	Resubstitution Error (%)	Cross-Validation Error (%)
Video #8	1,684	1,802	Latex foam	3.05	3.88
Video #9	725	954	Propane gas	4.09	6.08
Video #10	1,120	1,844	Wood crib	12.04	5.48
Total	**3,529**	**4,600**		**6.87**	**6.78**

A confusion matrix heat map is used to quantitatively visualize the performance of the proposed method, as illustrated in Figure 32 and Figure 33. As seen on the right, as the value of the heat map increases, the representing color becomes darker red and as the value lowers, the color becomes lighter pink. The five predicted classes are represented on the x-axis and the five actual classes are represented on the y-axis. All correct predictions are located on the diagonal of the matrix while wrong predictions are off-diagonal.

As seen in Figure 32, the proposed method has high accuracy in predictions with the training dataset when distinguishing smoke and fire as well as classifying fire and fire reflection. However, there is a higher error rate when classifying other objects, smoke and smoke reflection. The reason for such results is due to the tests with small amounts of smoke and low temperatures that cause the proposed method to produce erroneous results during feature recognition and classification. When there is a small amount of smoke present, texture patterns are diminished and the intensity is too low to distinguish smoke from other objects. The data in Figure 33 indicates that the proposed method has high accuracy in classification of the test dataset as well. Similar to the training dataset results, the proposed method produces high accuracy results for smoke and fire, fire and fire reflection but lower accuracy for other objects, smoke and smoke reflection.

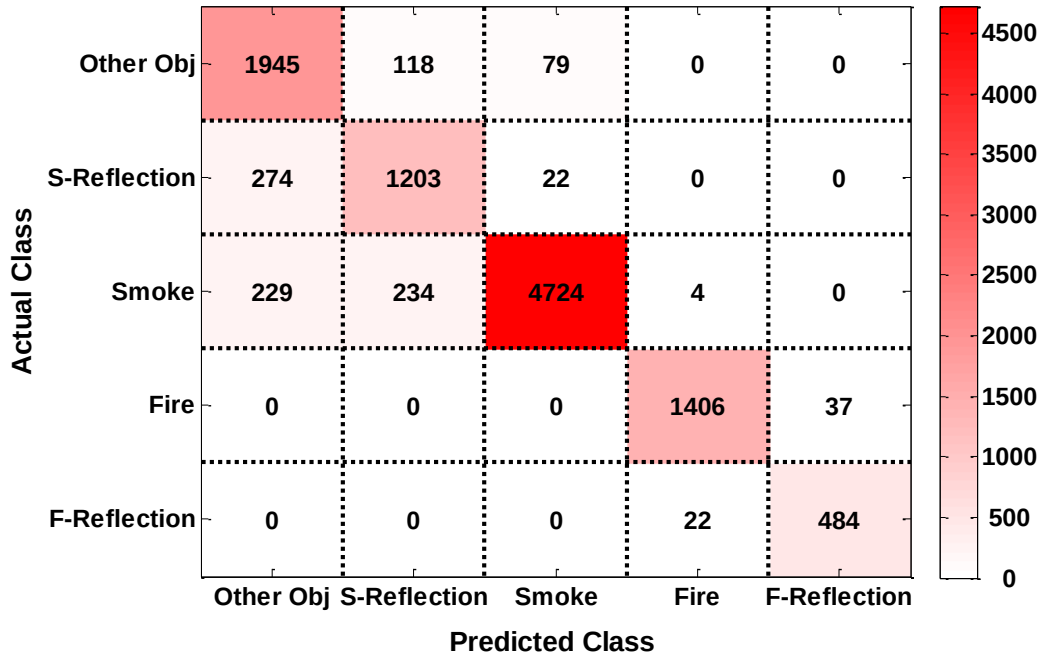


Figure 32. Confusion matrix heat map of the training dataset results.

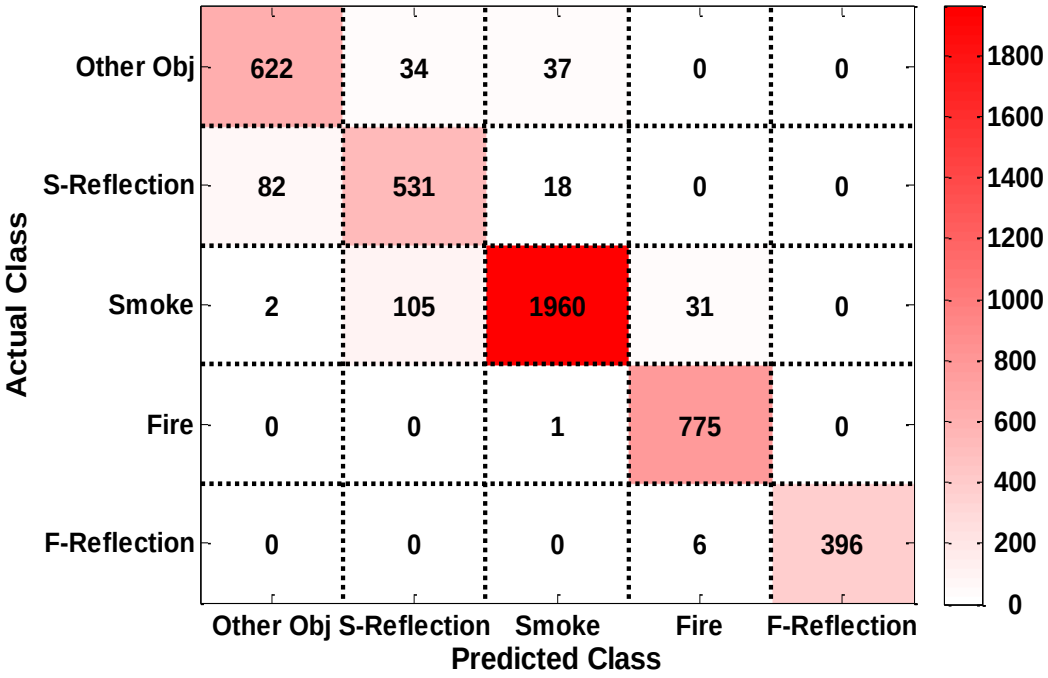


Figure 33. Confusion matrix heat map of the test dataset results

The performance of the proposed method was also measured through quantitatively analyzing the relevance between classes by using confusion matrix in Figure 32 and Figure 33 to assess four types of measurements; recall, precision, F-measure and G-measure. Unlike binary classification, when three or more classes are used, recall and precision can properly and quantitatively estimate the relevance between classes. Recall measures the fraction of positive examples from the positive group of the actual class and precision measures the fraction of positive instances from the group that the classifier predicted to be positive [102]. F-measure is the harmonic mean and G-measure is the geometric mean of recall and precision [123] and are calculated using the following equations

$$F - measure = \frac{2(recall \cdot precision)}{recall + precision} \tag{18}$$

$$G - measure = \sqrt{recall \cdot precision} \tag{19}$$

Figure 34 and Figure 35 show the quantitative analysis of classes versus each other for training and test datasets, where the x-axis contains the five classes and y-axis is the percentage of each of the four accuracy measures. For the training dataset results in Figure 34, smoke reflection and other objects have the lowest classification accuracy of the five classes, with percentages of the different measures ranging from 77-90%. Smoke reflection has the lowest results due to misclassifications between other objects and smoke. However, the classes that need to be identified for autonomous location of a fire (i.e., fire, smoke) have the highest accuracy of classification with measures ranging from 91-97%. The test dataset results given in Figure 35 contain similar trends to the training dataset except the accuracies are higher. In the test dataset, smoke and fire classes are classified with 93-99.9% accuracy. These results further demonstrate the method can accurately classify the classes needed to determine the robot heading.

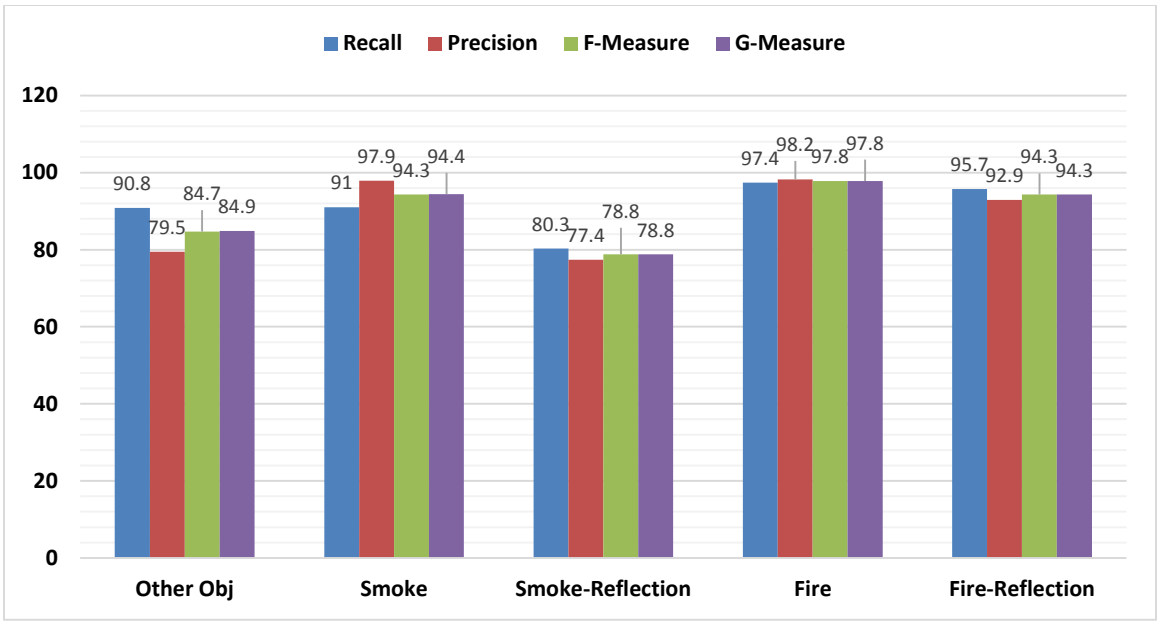


Figure 34. Training dataset results of recall, precision, F-measure and G-measure.

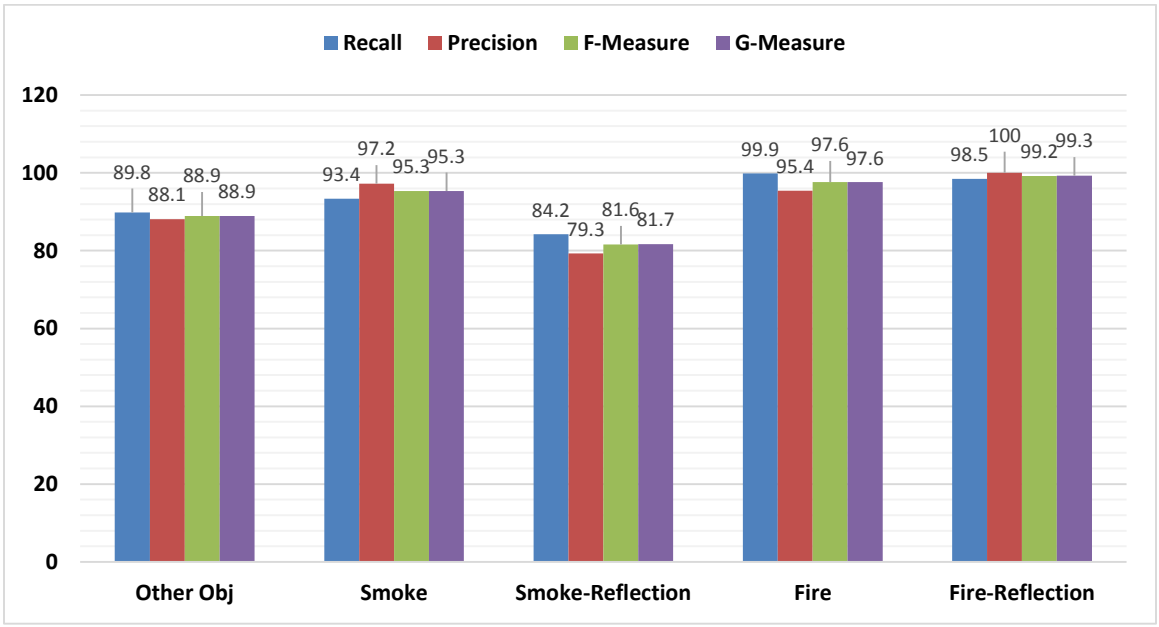


Figure 35. Test dataset results of recall, precision, F-measure and G-measure

5.9 Conclusion

A real-time probabilistic classification method was developed for fire, smoke, their thermal reflections and other objects using infrared images. This algorithm was developed for use on a robot that will autonomously locate fires inside of a structure where the fire is outside the robot field of view. Thermal images were used to extract features due to the fact that long wavelength IR is capable of imaging through zero visibility environments. An accurate robot heading requires that the robot be able to differentiate between desired characteristics, such as fire and smoke, and those that may lead the robot in the incorrect direction, such as thermal reflections and other hot objects.

The probabilistic classification method in this paper provides a robust, real-time algorithm that uses thermal images to classify fire and smoke with high accuracy. This method involved extracting candidates from 14-bit thermal images streamed from an infrared camera and analyzing four statistical texture features to characterize and classify the candidates. By calculating the posterior probability based on Bayes' theorem, the candidates were classified by the maximum priority decision rule.

Large-scale fire test video was used with supervised machine learning to produce over 76,000 data points to both train and validate the algorithm. Through analysis of this data, the most relevant first and second order statistics texture features were identified to be mean and variance of the intensity as well as entropy and inverse difference moment. Lower classification accuracy was measured for smaller fires (less than 25 kW) that have lighter smoke and lower temperatures. Overall, the classification performance error was measured to be 6.8% based on validation using the test dataset not included in the original training dataset. In addition, the precision, recall, F-measure, and G-measure were 93.5 – 99.9% for classifying fire and smoke using the test dataset, which are needed in the algorithm for leading the robot toward the fire.

6 Conclusions

6.1 Summary

In order to reduce firefighter deaths and increase their effectiveness on the job, this work has presented autonomous navigation, perception, and probability based fire location for intelligent firefighting robot. A real-time local obstacle avoidance was developed to avoid complex static and unexpected dynamic obstacles while computing a safe heading. Sensor fusion based locating fire algorithm was also developed to navigate toward a fire outside of the robot field of view while avoiding obstacles. A long-waved infrared camera FLIR A35 was used to analyze local conditions in an unknown fire smoke environment and probabilistically classify fire, smoke, and their thermal reflections to provide firefighting robots with a proper heading.

In Chapter 2, the weighted vector method (WVM) was presented as a real-time local obstacle avoidance method for autonomous robots in indoor environments where there may be dynamic obstacles and goals. WVM uses the vector representation to perceive obstacles of various shapes. To reduce computation to allow real-time implementation, WVM divides the configuration space into four areas and distinguishes necessary and unnecessary obstacle information in local environments. Based on deployment information of necessary obstacles, WVM uses five modes that are determined by a decision tree. Each mode has different weighting factors for obstacle avoidance. The weighted vectors in each mode are used to calculate safe headings for robot movement. Furthermore, WVM integrates goals and sub-goals for obstacle avoidance and overcomes problems in local obstacle avoidance.

In Chapter 3, a local dynamic goal-based fire location was presented, which fuses long wavelength infrared camera, ultraviolet radiation sensor, and LIDAR to efficiently locate a fire. WVM in Chapter 2 was applied for obstacle avoidance while moving toward fire outside of the robot field of view. For its validation, an experimental test-bed was constructed with a hallway and two rooms, with one of the rooms containing a real size fire created by propane gas. The robot immediately calculates its path towards the fire, moves towards it avoiding obstacles, and

ultimately finds the fire. When the fire is out, the robot returns to its original starting place. But there were false headings due to just using thresholding based on fire features in the FOV.

In Chapter 4, appropriate combination of features was analyzed to accurately classify fire, smoke, their thermal reflection and other hot objects using thermal images. The 14-bit gray-scale images from a single infrared camera were used to extract motion and texture features by applying a clustering-based auto-thresholding technique. Bayesian classification is performed to probabilistically identify multiple classes during real-time implementation. To find the best combination of features, the multi-objective optimization was implemented using resubstitution and cross-validation errors as objective functions. Large-scale fire tests with different fire sources were conducted to create a range of temperature and smoke conditions. Among the twelve motion and texture features, mean intensity (MNI), variance (VAR), entropy (ENT) and inverse difference moment (IDM) were selected producing the highest performance in the classification.

In Chapter 5, a real-time probabilistic classification method was presented, which can classify fire, smoke, their thermal reflections, and other hot objects. This classification enables the robot to differentiate between desired characteristics, such as fire and smoke, and those that may lead the robot in the incorrect direction, such as thermal reflections and other hot objects. The probabilistic classification method autonomously locates a fire inside of a structure where the fire is outside the robot field of view. Thermal images were used to extract features due to the fact that long wavelength IR is capable of imaging through zero visibility environments. Large-scale fire test video was used with supervised machine learning to produce over 76,000 data points to both train and validate the algorithm. Through analysis of this data in Chapter 4, the most relevant features were identified. Overall, the classification performance error was measured to be 6.8% based on validation using the test dataset not included in the original training dataset. In addition, the precision, recall, F-measure, and G-measure were 93.5 – 99.9% for classifying fire and smoke using the test dataset, which are needed in the algorithm for leading the robot toward the fire.

6.2 Recommendations

The WVM in Chapter 2 used two-dimensional obstacle information for autonomous obstacle avoidance of a mobile robot. By using a Lidar rotating to collect obstacles in three dimensional coordinates, WVM can be extended to three-dimensional obstacle avoidance for a humanoid robot. This can function in avoiding unexpected dynamic obstacles while moving toward a fire as well as calculating safe headings.

Another is to search other features e.g. skewness, kurtosis, etc. that can improve the accuracy of the classification. In Chapter 4, twelve motion and texture features were studied for the classification of fire, smoke, thermal reflections and other hot objects, and a combination of four features that has the highest accuracy of the classification were selected. However, there may exist features that are not included in this research but can improve the classification. Moreover, a new single feature can be created by integrating three or higher order statistical textures that can more accurately identify fire, smoke thermal reflections and other hot objects without using multiple features.

The other is to improve a heading direction under smoke-filled situation by probabilistically fusing data from both optical flow and hottest temperature in the scene. The direction of smoke flow and the hottest spots can be estimated to determine a more accurate and promising direction toward fire source. By integrating 3-D mapping of fire environments by 3-D SLAM, fire location can be more accurately estimated and displayed for the robot itself and operators. In addition, the classification method introduced in Chapter 5 can be applied to fire-related probabilistic frameworks including water sprays, brands, and combustible materials. Object and feature extractions and classification method similar to this study can be utilized. However different features should be considered for these type of characterization.

References

[1] NFPA, *Deadliest Fires in the U.S. with 5 or more Firefighter Deaths at the Fire Grounds, 1977-2012*: Fire Incident Data Organization, 2013.

[2] M. J. Karter and J. L. Molis, *US Firefighter Injuries-2011*: National Fire Protection Association, Fire Analysis and Research Division, 2012.

[3] R. F. Fahy, P. R. LeBlanc, and J. L. Molis, *Firefighter Fatalities in the United States-2011*: NFPA, 2012.

[4] M. J. Karter, *Fire loss in the United States during 2010*: National Fire Protection Association Quincy, MA, 2011.

[5] M. J. Karter, "Selected Special Analyses of Firefighter Fatalities," National Fire Protection Association Quincy, MA2005.

[6] E. Ronchi, S. Gwynne, D. Purser, and P. Colonna, "Representation of the impact of smoke on agent walking speeds in evacuation models," *Fire technology,* vol. 49, pp. 411-431, 2013.

[7] J. L. Bryan, "Behavioral response to fire and smoke," *SFPE handbook of fire protection engineering,* vol. 2, 2002.

[8] M. Wright, G. Cook, and G. Webber, "The effects of smoke on people's walking Speeds using overhead lighting and Wayguidance provision," in *Proceedings of the 2nd international symposium on human behaviour in fire. MIT, Boston,* 2001, pp. 275-284.

[9] T. Jin, "Visibility through fire smoke (No. 42): report of Fire Research Institute of Japan," 1976.

[10] C. Bertram, M. H. Evans, M. Javaid, T. Stafford, and T. Prescott, "Sensory augmentation with distal touch: the tactile helmet project," in *Biomimetic and Biohybrid Systems*, ed: Springer, 2013, pp. 24-35.

[11] P. Rutherford, "Auditory Navigation and the Escape from Smoke Filled Buildings," in *CAAD futures 1997*, ed: Springer, 1997, pp. 299-304.

[12] K. Miyazawa, "Fire robots developed by the Tokyo Fire Department," *Advanced Robotics,* vol. 16, pp. 553-556, 2002.

[13] C. F. Tan, M. Alkahari, and A. Rahman, "Development of Ground Vehicle for Fire Fighting Purpose," 2011.

[14] T. N. Khoon, P. Sebastian, and A. B. S. Saman, "Autonomous Fire Fighting Mobile Platform," *Procedia Engineering*, vol. 41, pp. 1145-1153, 2012.

[15] P. Liljeback, O. Stavdahl, and A. Beitnes, "SnakeFighter-development of a water hydraulic fire fighting snake robot," in *Control, Automation, Robotics and Vision, 2006. ICARCV'06. 9th International Conference on*, 2006, pp. 1-6.

[16] J. H. Hong, B.-C. Min, J. M. Taylor, V. Raskin, and E. T. Matson, "NL-based communication with firefighting robots," in *Systems, Man, and Cybernetics (SMC), 2012 IEEE International Conference on*, 2012, pp. 1461-1466.

[17] P. H. Chang, Y. H. Kang, G. R. Cho, J. H. Kim, M. Jin, J. Lee, *et al.*, "Control architecture design for a fire searching robot using task oriented design methodology," in *SICE-ICASE, 2006. International Joint Conference*, 2006, pp. 3126-3131.

[18] Y. D. Kim, Y. G. Kim, S. H. Lee, J. H. Kang, and J. An, "Portable fire evacuation guide robot system," in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, 2009, pp. 2789-2794.

[19] D. Longo and G. Muscato, "CLAWAR WP3 Applications: Natural/Outdoor and Underwater Robots," in *Climbing and Walking Robots*, ed: Springer, 2005, pp. 1159-1170.

[20] J. Penders, L. Alboul, U. Witkowski, A. Naghsh, J. Saez-Pons, S. Herbrechtsmeier, *et al.*, "A robot swarm assisting a human fire-fighter," *Advanced Robotics*, vol. 25, pp. 93-117, 2011.

[21] J. Suthakorn, S. Shah, S. Jantarajit, W. Onprasert, W. Saensupo, S. Saeung, *et al.*, "On the design and development of a rough terrain robot for rescue missions," in *Robotics and Biomimetics, 2008. ROBIO 2008. IEEE International Conference on*, 2009, pp. 1830-1835.

[22] J.-H. Kim, J. Starr, and B. Lattimer, "Firefighting Robot Stereo Infrared Vision and Radar Sensor Fusion for Imaging through Smoke," *Fire Technology*, pp. 1-23, 2014/06/05 2014.

[23] J. W. Starr and B. Y. Lattimer, "Application of Thermal Infrared Stereo Vision in Fire Environments," presented at the IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM 2013), Wollongong, Austrailia, 2013.

[24] J.-H. Kim, B. Keller, and B. Y. Lattimer, "Sensor fusion based seek-and-find fire algorithm for intelligent firefighting robot," in *Advanced Intelligent Mechatronics (AIM), 2013 IEEE/ASME International Conference on*, 2013, pp. 1482-1486.

[25] J. W. Starr and B. Lattimer, "Evaluation of Navigation Sensors in Fire Smoke Environments," *Fire Technology*, pp. 1-23, 2012.

[26] J. W. Starr and B. Lattimer, "A comparison of IR stereo vision and LIDAR for use in fire environments," in *Sensors, 2012 IEEE*, 2012, pp. 1-4.

[27] J. G. McNeil, J. Starr, and B. Y. Lattimer, "Autonomous Fire Suppression Using Multispectral Sensors," presented at the IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM 2013), Wollongong, Austrailia, 2013.

[28] S. Kleindienst, A. Dimmeler, and D. Clement, "Obscuration effects of smoke clouds," in *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, 1995, pp. 440-451.

[29] U. Koylu and G. Faeth, "Spectral extinction coefficients of soot aggregates from turbulent diffusion flames," *Transactions of the ASME-C-Journal of Heat Transfer*, vol. 118, pp. 415-421, 1996.

[30] W. Dalzell and A. Sarofim, "Optical constants of soot and their application to heat-flux calculations," *J. Heat Transfer*, vol. 91, pp. 100-104, 1969.

[31] T. Jin, "Chapter 2-4 Visibility and Human Behavior in Fire Smoke," in *SFPE Handbook of Fire Protection Engineering, 4th Ed*, ed: The Society of Fire Protection Engineers, 2008.

[32] R. Felicetti, "Assessment methods of fire damages in concrete tunnel linings," *Fire technology*, vol. 49, pp. 509-529, 2013.

[33] M. Hajpál, "Changes in sandstones of historical monuments exposed to fire or high temperature," *Fire Technology*, vol. 38, pp. 373-382, 2002.

[34] L. Cider, "Cleaning and reliability of smoke-contaminated electronics," *Fire Technology*, vol. 29, pp. 226-245, 1993.

[35] S. Quinlan and O. Khatib, "Elastic bands: Connecting path planning and control," in *Robotics and Automation, 1993. Proceedings., 1993 IEEE International Conference on*, 1993, pp. 802-807.

[36] V. Sezer and M. Gokasan, "A novel obstacle avoidance algorithm:“Follow the Gap Method”," *Robotics and Autonomous Systems,* 2012.

[37] C.-P. Wu, T.-T. Lee, and C.-R. Tsai, "Obstacle avoidance motion planning for mobile robots in a dynamic environment with moving obstacles," *Robotica,* vol. 15, pp. 493-510, 1997.

[38] R. Simmons, "The curvature-velocity method for local obstacle avoidance," in *Robotics and Automation, 1996. Proceedings., 1996 IEEE International Conference on*, 1996, pp. 3375-3382.

[39] C. Shi, Y. Wang, and J. Yang, "A local obstacle avoidance method for mobile robots in partially known environment," *Robotics and Autonomous Systems,* vol. 58, pp. 425-434, 2010.

[40] J.-C. Latombe, *ROBOT MOTION PLANNING.: Edition en anglais*: Springer, 1990.

[41] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische mathematik,* vol. 1, pp. 269-271, 1959.

[42] A. Chakravarthy and D. Ghose, "Obstacle avoidance in a dynamic environment: A collision cone approach," *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on,* vol. 28, pp. 562-574, 1998.

[43] K. Fujimura, "Constrained Motion," in *Motion Planning in Dynamic Environments*, ed: Springer, 1991, pp. 127-142.

[44] V. Lumelsky and T. Skewis, "Incorporating range sensing in the robot navigation function," *Systems, Man and Cybernetics, IEEE Transactions on,* vol. 20, pp. 1058-1069, 1990.

[45] V. J. Lumelsky and A. A. Stepanov, "Path-planning strategies for a point mobile automaton moving amidst unknown obstacles of arbitrary shape," *Algorithmica,* vol. 2, pp. 403-430, 1987.

[46] Y. Koren and J. Borenstein, "Potential field methods and their inherent limitations for mobile robot navigation," in *Robotics and Automation, 1991. Proceedings., 1991 IEEE International Conference on*, 1991, pp. 1398-1404.

[47] M. G. Park, J. H. Jeon, and M. C. Lee, "Obstacle avoidance for mobile robots using artificial potential field approach with simulated annealing," in *Industrial Electronics, 2001. Proceedings. ISIE 2001. IEEE International Symposium on*, 2001, pp. 1530-1535.

[48] N. Y. Ko and R. G. Simmons, "The lane-curvature method for local obstacle avoidance," in *Intelligent Robots and Systems, 1998. Proceedings., 1998 IEEE/RSJ International Conference on*, 1998, pp. 1615-1621.

[49] J. Borenstein and Y. Koren, "The vector field histogram-fast obstacle avoidance for mobile robots," *Robotics and Automation, IEEE Transactions on,* vol. 7, pp. 278-288, 1991.

[50] I. Ulrich and J. Borenstein, "VFH+: Reliable obstacle avoidance for fast mobile robots," in *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on*, 1998, pp. 1572-1577.

[51] R. Siegwart and I. R. Nourbakhsh, *Introduction to Autonomous Mobile Robotos*: The MIT press, 2004.

[52] L. McFetridge and M. Ibrahim, "A new methodology of mobile robot navigation: The agoraphilic algorithm," *Robotics and Computer-Integrated Manufacturing,* vol. 25, pp. 545-551, 2009.

[53] H.-P. Huang and P.-C. Lee, "Real-time algorithm for obstacle avoidance of autonomous mobile robots," *Robotica,* vol. 10, pp. 217-227, 1992.

[54] I. Ulrich and J. Borenstein, "VFH*: local obstacle avoidance with look-ahead verification," in *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, 2000, pp. 2505-2511.

[55] K. Ioannidis, G. C. Sirakoulis, and I. Andreadis, "Cellular ants: A method to create collision free trajectories for a cooperative robot team," *Robotics and Autonomous Systems,* vol. 59, pp. 113-127, 2011.

[56] S. S. Chiddarwar and N. Ramesh Babu, "Conflict free coordinated path planning for multiple robots using a dynamic path modification sequence," *Robotics and Autonomous Systems,* vol. 59, pp. 508-518, 2011.

[57] S. Lahouar, E. Ottaviano, S. Zeghoul, L. Romdhane, and M. Ceccarelli, "Collision free path-planning for cable-driven parallel robots," *Robotics and Autonomous Systems,* vol. 57, pp. 1083-1093, 2009.

[58] M. Galicki, "Collision-free control of an omni-directional vehicle," *Robotics and Autonomous Systems,* vol. 57, pp. 889-900, 2009.

[59] G. Klančar, D. Matko, and S. Blažič, "A control strategy for platoons of differential drive wheeled mobile robot," *Robotics and Autonomous Systems,* vol. 59, pp. 57-64, 2011.

[60] G. Cook, *Mobile robots: navigation, control and remote sensing*: John Wiley & Sons, 2011.

[61] M. Kim, K. Cho, B.-J. You, and C.-W. Lee, "Task planning for humanoid robots using look-up table," *Robotics and Autonomous Systems,* vol. 40, pp. 205-212, 2002.

[62] R. C. Luo and K. L. Su, "Autonomous fire-detection system using adaptive sensory fusion for intelligent security robot," *Mechatronics, IEEE/ASME Transactions on,* vol. 12, pp. 274-281, 2007.

[63] J. A. Milke, M. E. Hulcher, C. L. Worrell, D. T. Gottuk, and F. W. Williams, "Investigation of multi-sensor algorithms for fire detection," *Fire technology,* vol. 39, pp. 363-382, 2003.

[64] H.-C. Kuo and H. Chang, "A real-time shipboard fire-detection system based on grey-fuzzy algorithms," *Fire Safety Journal,* vol. 38, pp. 341-363, 2003.

[65] D. T. Gottuk, S. A. Hill, C. F. Schemel, and B. D. Strehlen, "Identification of fire signatures for shipboard multi-criteria fire detection systems," DTIC Document1999.

[66] M. A. Jackson and I. Robins, "Gas sensing for fire detection: Measurements of CO, CO2, H2, O2, and smoke density in European standard fire tests," *Fire Safety Journal,* vol. 22, pp. 181-205, // 1994.

[67] B. U. Töreyin, R. G. Cinbiş, Y. Dedeoğlu, and A. E. Çetin, "Fire detection in infrared video using wavelet analysis," *Optical Engineering,* vol. 46, pp. 067204-067204-9, 2007.

[68] Y. Dedeoglu, B. U. Töreyin, U. Güdükbay, and A. E. Cetin, "Real-Time Fire and Flame Detection in Video," in *ICASSP (2)*, 2005, pp. 669-672.

[69] T. Celik, H. Demirel, H. Ozkaramanli, and M. Uyguroglu, "Fire detection using statistical color model in video sequences," *Journal of Visual Communication and Image Representation,* vol. 18, pp. 176-185, 2007.

[70] B. U. Toreyin and A. E. Cetin, "Online detection of fire in video," in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, 2007, pp. 1-5.

[71] C.-C. Ho, "Machine vision-based real-time early flame and smoke detection," *Measurement Science and Technology,* vol. 20, p. 045502, 2009.

[72] D. Krstinić, D. Stipaničev, and T. Jakovčević, "Histogram-based smoke segmentation in forest fire detection system," *Information Technology and Control,* vol. 38, pp. 237-244, 2009.

[73] B. Ko, K.-H. Cheong, and J.-Y. Nam, "Early fire detection algorithm based on irregular patterns of flames and hierarchical Bayesian Networks," *Fire Safety Journal,* vol. 45, pp. 262-270, 2010.

[74] Y. Wang, T. W. Chua, R. Chang, and N. T. Pham, "Real-time smoke detection using texture and color features," in *Pattern Recognition (ICPR), 2012 21st International Conference on*, 2012, pp. 1727-1730.

[75] G. Marbach, M. Loepfe, and T. Brupbacher, "An image processing technique for fire detection in video images," *Fire safety journal,* vol. 41, pp. 285-289, 2006.

[76] J. Yang, F. Chen, and W. Zhang, "Visual-based smoke detection using support vector machine," in *Natural Computation, 2008. ICNC'08. Fourth International Conference on*, 2008, pp. 301-305.

[77] H. Maruta, Y. Kato, A. Nakamura, and F. Kurokawa, "Smoke detection in open areas using its texture features and time series properties," in *Industrial Electronics, 2009. ISIE 2009. IEEE International Symposium on*, 2009, pp. 1904-1908.

[78] M. I. Chacon-Murguia and F. J. Perez-Vargas, "Thermal video analysis for fire detection using shape regularity and intensity saturation features," in *Pattern Recognition*, ed: Springer, 2011, pp. 118-126.

[79] T. X. Tung and J.-M. Kim, "An effective four-stage smoke-detection algorithm using video images for early fire-alarm systems," *Fire Safety Journal,* vol. 46, pp. 276-282, 2011.

[80] S. Verstockt, S. Van Hoecke, T. Beji, B. Merci, B. Gouverneur, A. E. Cetin, *et al.*, "A multi-modal video analysis approach for car park fire detection," *Fire Safety Journal,* vol. 57, pp. 44-57, 2013.

[81] W. Phillips Iii, M. Shah, and N. da Vitoria Lobo, "Flame recognition in video," *Pattern recognition letters,* vol. 23, pp. 319-327, 2002.

[82] T.-H. Chen, P.-H. Wu, and Y.-C. Chiou, "An early fire-detection method based on image processing," in *Image Processing, 2004. ICIP'04. 2004 International Conference on*, 2004, pp. 1707-1710.

[83] D. Han and B. Lee, "Development of early tunnel fire detection algorithm using the image processing," in *Advances in Visual Computing*, ed: Springer, 2006, pp. 39-48.

[84] K.-H. Cheong, B.-C. Ko, and J.-Y. Nam, "Automatic fire detection system using CCD camera and Bayesian network," in *Electronic Imaging 2008*, 2008, pp. 68130S-68130S-12.

[85] D. Han and B. Lee, "Flame and smoke detection method for early real-time detection of a tunnel fire," *Fire Safety Journal*, vol. 44, pp. 951-961, 2009.

[86] B. C. Ko, K.-H. Cheong, and J.-Y. Nam, "Fire detection based on vision sensor and support vector machines," *Fire Safety Journal*, vol. 44, pp. 322-329, 2009.

[87] Y. Chunyu, F. Jun, W. Jinjun, and Z. Yongming, "Video fire smoke detection using motion and color features," *Fire technology*, vol. 46, pp. 651-663, 2010.

[88] F. Yuan, "Video-based smoke detection with histogram sequence of LBP and LBPV pyramids," *Fire safety journal*, vol. 46, pp. 132-139, 2011.

[89] F. Lafarge, X. Descombes, and J. Zerubia, "Textural kernel for SVM classification in remote sensing: Application to forest fire detection and urban area extraction," in *Image Processing, 2005. ICIP 2005. IEEE International Conference on*, 2005, pp. III-1096-9.

[90] F. Amon, V. Benetis, J. Kim, and A. Hamins, "Development of a performance evaluation facility for fire fighting thermal imagers," in *Defense and Security*, 2004, pp. 244-252.

[91] F. Amon, N. Bryner, and A. Hamins, "Evaluation of thermal imaging cameras used in fire fighting applications," in *Defense and Security*, 2004, pp. 44-53.

[92] F. Amon and A. Ducharme, "Image Frequency Analysis for Testing of Fire Service Thermal Imaging Cameras," *Fire technology*, vol. 45, pp. 313-322, 2009.

[93] F. D. Maxwell, "A portable IR system for observing fire thru smoke," *Fire Technology*, vol. 7, pp. 321-331, 1971.

[94] A. Bruhn, J. Weickert, and C. Schnörr, "Lucas/Kanade meets Horn/Schunck: Combining local and global optic flow methods," *International Journal of Computer Vision*, vol. 61, pp. 211-231, 2005.

[95] B. Galvin, B. McCane, K. Novins, D. Mason, and S. Mills, "Recovering Motion Fields: An Evaluation of Eight Optical Flow Algorithms," in *BMVC*, 1998, pp. 195-204.

[96] J. L. Barron, D. J. Fleet, and S. S. Beauchemin, "Performance of optical flow techniques," *International journal of computer vision*, vol. 12, pp. 43-77, 1994.

[97] R. M. Haralick, K. Shanmugam, and I. H. Dinstein, "Textural features for image classification," *Systems, Man and Cybernetics, IEEE Transactions on,* pp. 610-621, 1973.

[98] O. Barnich and M. Van Droogenbroeck, "ViBe: A universal background subtraction algorithm for video sequences," *Image Processing, IEEE Transactions on,* vol. 20, pp. 1709-1724, 2011.

[99] R. Maini and H. Aggarwal, "Study and comparison of various image edge detection techniques," *International Journal of Image Processing (IJIP),* vol. 3, pp. 1-11, 2009.

[100] N. Otsu, "A threshold selection method from gray-level histograms," *Automatica,* vol. 11, pp. 23-27, 1975.

[101] B. Scholkopft and K.-R. Mullert, "Fisher discriminant analysis with kernels," 1999.

[102] P. Harrington, *Machine Learning in Action*: Manning Publications Co., 2012.

[103] S. B. Kotsiantis, "Supervised machine learning: a review of classification techniques," *Informatica (03505596),* vol. 31, 2007.

[104] F. Van Der Heijden, R. Duin, D. De Ridder, and D. M. Tax, *Classification, parameter estimation and state estimation: an engineering approach using MATLAB*: John Wiley & Sons, 2005.

[105] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *IJCAI*, 1995, pp. 1137-1145.

[106] K. Deb, *Multi-objective optimization using evolutionary algorithms* vol. 16: John Wiley & Sons, 2001.

[107] D. J. Pack, R. Avanzato, D. J. Ahlgren, and I. M. Verner, "Fire-fighting mobile robotics and interdisciplinary design-comparative perspectives," *Education, IEEE Transactions on,* vol. 47, pp. 369-376, 2004.

[108] S. Dearie, K. Fisher, B. Rajala, and S. Wasson, "Design and construction of a fully autonomous fire fighting robot," in *Electrical Insulation Conference and Electrical Manufacturing & Coil Winding Conference, 2001. Proceedings*, 2001, pp. 303-310.

[109] B. U. Töreyin, Y. Dedeoğlu, U. Güdükbay, and A. E. Cetin, "Computer vision based method for real-time fire and flame detection," *Pattern recognition letters,* vol. 27, pp. 49-58, 2006.

[110] T.-H. Chen, Y.-H. Yin, S.-F. Huang, and Y.-T. Ye, "The smoke detection for early fire-alarming system base on video processing," in *Intelligent Information Hiding and Multimedia Signal Processing, 2006. IIH-MSP'06. International Conference on*, 2006.

[111] B. Ni, G. Wang, and P. Moulin, "Rgbd-hudaact: A color-depth video database for human daily activity recognition," in *Consumer Depth Cameras for Computer Vision*, ed: Springer, 2013, pp. 193-208.

[112] J. R. Bach, C. E. Fuller, A. Gupta, B. Horowitz, R. Jain, and C.-f. Shu, "Visual image database search engine which allows for different schema," ed: Google Patents, 1999.

[113] T. Kato, T. Kurita, N. Otsu, and K. Hirata, "A sketch retrieval method for full color image database-query by visual example," in *Pattern Recognition, 1992. Vol. I. Conference A: Computer Vision and Applications, Proceedings., 11th IAPR International Conference on*, 1992, pp. 530-533.

[114] S. M. Pizer, E. P. Amburn, J. D. Austin, R. Cromartie, A. Geselowitz, T. Greer, *et al.*, "Adaptive histogram equalization and its variations," *Computer vision, graphics, and image processing*, vol. 39, pp. 355-368, 1987.

[115] A. El Maadi and X. Maldague, "Outdoor infrared video surveillance: A novel dynamic technique for the subtraction of a changing background of IR images," *Infrared physics & technology*, vol. 49, pp. 261-265, 2007.

[116] M. K. Tsatsanis and G. B. Giannakis, "Object and texture classification using higher order statistics," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, pp. 733-750, 1992.

[117] P. Mather and B. Tso, *Classification methods for remotely sensed data*: CRC press, 2003.

[118] D. A. Clausi, "An analysis of co-occurrence texture statistics as a function of grey level quantization," *Canadian Journal of remote sensing*, vol. 28, pp. 45-62, 2002.

[119] A. Huda and S. Taib, "Suitable features selection for monitoring thermal condition of electrical equipment using infrared thermography," *Infrared Physics & Technology*, vol. 61, pp. 184-191, 2013.

[120] O. Kramer, "K-Nearest Neighbors," in *Dimensionality Reduction with Unsupervised Nearest Neighbors*, ed: Springer, 2013, pp. 13-23.

[121] J. R. Quinlan, "Induction of decision trees," *Machine learning*, vol. 1, pp. 81-106, 1986.

[122] D. J. Hand, H. Mannila, and P. Smyth, *Principles of data mining*: MIT press, 2001.

[123] D. M. Powers, "Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation," 2011.

Appendix A: Permissions

01_01

Springer

PERMISSION LETTER

August 8, 2014

Springer reference

Fire Technology (2014)
Firefighting Robot Stereo Infrared Vision and Radar Sensor Fusion for Imaging through Smoke
Jong-Hwan Kim, Joseph W. Starr, Brian Y. Lattimer
Table 1 Firefighting robots
Excerpts from page 1, 2, 3, and 4
ISSN: 0015-2684
DOI: 10.1007/s10694-014-0413-6
Journal no.: 10694

Your project

Requestor: Jong Hwan, Kim
hspkorea@vt.edu
Virginia Tech
Mechanical Engineering Department
Extreme Environments, Robotics, and Materials (ExtReMe) Lab
http://www.me.vt.edu/extreme/
Rm. 406, SEB
635 Prices Fork Road
Blacksburg, VA 24061
Publisher: Virginia Tech
Title: Autonomous Navigation, Perception and Probabilistic Fire Location for an Intelligent Firefighting Robot
Language: English
Territory: Worldwide

With reference to your request to reuse material in which **Springer Science+Business Media** controls the copyright, our permission is granted free of charge under the following conditions:

Springer material

- represents original material which does not carry references to other sources. If the material in question appears with a credit to another source, this permission is not valid and authorization has to be obtained from the original copyright holder;
- requires full credit (Springer and the original publisher, book/journal title, chapter/article title, volume, year of publication, page, name(s) of author(s), original copyright notice) to the publication in which the material was originally published by adding: "With kind permission of Springer Science+Business Media";
- figures, illustrations, and tables may be altered minimally to serve your work. Any other abbreviations, additions, deletions and/or any other alterations shall be made only with prior written authorization of the author and/or Springer Science+Business Media;
- may not be republished in electronic open access;
- Springer does not supply original artwork or content.

This permission

- is non-exclusive;
- includes use in electronic form, provided it is password protected, on intranet, or CD-ROM/DVD, or E-book/E-journal;

Branch of Springer-Verlag GmbH, Heidelberger Platz 3, 14197 Berlin, Germany | Amtsgericht Berlin-Charlottenburg, HRB 91881 B
Managing Directors: Derk Haank, Martin Mos, Peter Hendriks | Springer is part of Springer Science+Business Media

Springer

PERMISSION LETTER

- is personal to you and may not be sublicensed, assigned, or transferred by you without Springer's written permission;
- explicitly excludes the right for derivatives.

Additional Provisions

- ***Warranties***
  Springer Science+Business Media makes no representations or warranties with respect to the licensed material.

- ***Indemnity***
  You hereby indemnify and agree to hold harmless Springer Science+Business Media and its respective officer, directors, employees, and agents, from and against any and all claims arising out of your use of the licensed material other than specifically authorized pursuant to this license.

Permission free of charge does not prejudice any rights we might have to charge for reproduction of our copyrighted material in the future.

This license is valid only when the conditions noted above are met.

Rights and Permissions
Springer Science+Business Media
Tiergartenstr. 17
69121 Heidelberg
Germany

Copyright Clearance Center RightsLink®

Home | Create Account | Help

IEEE Requesting permission to reuse content from an IEEE publication

Title: Sensor fusion based seek-and-find fire algorithm for intelligent firefighting robot

Conference Proceedings: Advanced Intelligent Mechatronics (AIM), 2013 IEEE/ASME International Conference on

Author: Jong-Hwan Kim; Keller, B.; Lattimer, B.Y.

Publisher: IEEE

Date: 9-12 July 2013

Copyright © 2013, IEEE

User ID

Password

Enable Auto Login

LOGIN

Forgot Password/User ID?

If you're a copyright.com user, you can login to RightsLink using your copyright.com credentials. Already **a RightsLink user** or want to learn more?

Thesis / Dissertation Reuse

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:

1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.
3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

BACK | CLOSE WINDOW

Copyright © 2014 Copyright Clearance Center, Inc. All Rights Reserved. Privacy statement.
Comments? We would like to hear from you. E-mail us at customercare@copyright.com