To facilitate rescue operations with the help of UAV and SLAM-algorithm

Mapping and imaging of an unknown area

Emma Sandgren & Thomas Henriksson

Department of Computer and Systems Sciences

Degree project 15 HE credits
Degree subject: Computer and Systems Sciences
Degree project at the bachelor level
Spring 2017
Advisor: Rahim Rahmani
Reviewer: Lars Asker
Swedish title: Att underlätta räddningsarbete med hjälp av drönare och SLAM-algoritm

Stockholms universitet

To facilitate rescue operations with the help of UAV and SLAM-algorithm

Mapping and imaging of an unknow area

Emma Sandgren & Thomas Henriksson

Abstract

During recent years, several large-scale accidents have happened. Every year different types of industrial accidents occur around the world. To facilitate rescue operations and gather information, technology that can access and map inaccessible areas is needed. To investigate possible solutions with the framework of design science, three different scenarios have been created in the robot simulation program V-REP, in which a drone equipped with laser scanner have driven through and mapped. The laser data have been processed with the algorithm BreezySlam, a SLAM-algorithm with the purpose of solving the problem of performing localisation and mapping at the same time. This resulted in three graphs that in two dimensions shows an imaging of respective area and the probable path of the drone. An evaluation has been performed and the conclusion drawn is that the artefact for a real-world setting is not reliable enough, but by limited resources still shows promising results.

Keywords

Mapping, imaging, drone, SLAM-algorithm, BreezySlam, laser scanner.

Sammanfattning

Under de senaste åren har flera storskaliga olyckor skett och varje år sker olika typer av industriella olyckor världen över. För att underlätta räddningsarbetet vid framtida olyckor krävs det teknik som dels kan ta sig fram i otillgängliga områden, dels kartlägga dessa för att ge kunskap om områdena. För att undersöka hur detta kan göras med design science som ramverk, har det i simuleringsprogrammet V-REP skapats tre olika scenarier som en drönare utrustad med en laserskanner åkt igenom och kartlagt. Laserdatan har sedan processats med hjälp av algoritmen BreezySlam, en SLAM-algoritm vars syfte är att lösa problemet med att samtidigt kunna utföra lokalisering och avbildning. Det har resulterat i tre grafer som visar, i två dimensioner, en avbildning av respektive omgivning och drönarens, av algoritmen, uppskattade väg. Dessa har utvärderats och slutsatsen som dras är att artefakten för en verklig situation inte är tillräckligt reliabel men utifrån begränsade resurser ändå visar på lovande resultat.

Synopsis form

Background	The need for support during rescue operations demands appropriate tools (drone equipped with laser scanner) and systems for mapping of area, to reduce risk of harm. Because the unreliability of GPS as a location device, something else must be used. This thesis has been made within the area of immersive networking.
Problem	Because of probable blocking of signals by collapsed objects and for a drone to map and navigate an area, another technique must be created and used, and the simultaneous localisation and mapping problem must be solved, which this thesis aims to do.
Research Question	How can automated drones be used with a purpose to map and graphically present, to humans, inaccessible and/or dangerous environments?
Method	The main research strategy used for this thesis is design science, which consists of five different steps. These are completed with help of relevant research (simulation) and data collection (observation) methods. The result is an artefact consisting of a simulation environment and a SLAM-algorithm.
Result	The result is based on data collected through demonstrating the implemented artefact, in the form of three images and the artefact itself. It doesn't quite fulfil the aim applied in a real-world scenario but shows promising result due to limited resources.
Discussion	Due to limited resources limitations regarding the artefacts application to a real-world setting exists. Since the artefact was made in a computer environment, the risk and possible harm has been minimal. It's possible with limited resources to create a system that maps and images an area, which can be useful with more research in a real-world situation.

Table of Contents

List of Figures

List of Tables

Table of Listings

1 Introduction

This chapter will give an understanding to the purpose of this thesis, to facilitate and make rescue work easier with the help of drones, and from which background the research question stems from. Furthermore, problems surrounding the use of GPS as a positioning method and why alternative techniques must be used will be presented.

1.1 Background

Ever since the 1980s have the number of industrial accidents around the world (Asia, Europe, Americas and Africa) increased, with a maximum during the early 2000 followed by a decline, see Figure 1 (CRED 2017).

Figure 1: Number of accidents in Asia, Americas, Europe & Africa since 1980 (CRED 2017).

During the period 1980 to 2016 there were in total 1288 number of industrial accidents, where the three most occurring types are explosions (50 %), fire (15 %) and collapses (11 %). The number of deaths caused by different types of accidents (in Asia, Europe, Americas and Africa) during the same period were 42 734, with explosion at the top (57 %) followed by collapse (13 %) and fire (10 %). Despite the decline in the amount of industrial accidents since the spike in the early 2000s have the number of deaths stayed at a relatively even level, see Figure 2. (ibid.)

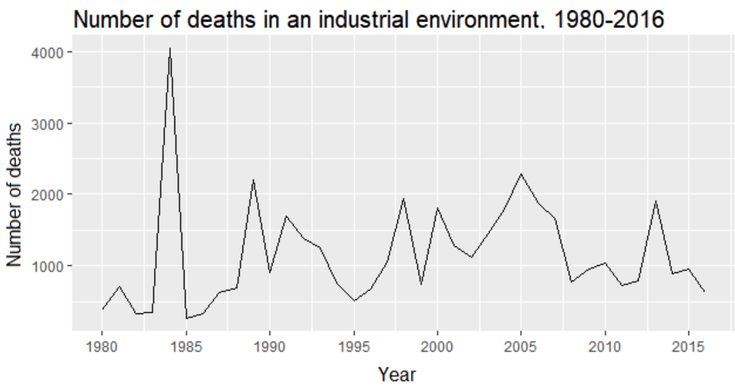


Figure 2: Number of deaths caused by industrial accidents in Africa, America, Europe & Asia since 1980 (CRED 2017).

An example of a larger scale accident happened 2011 in Fukushima, where a reactor in a nuclear power plant dispersed radiation from radioactive material (Nationalencyklopedin 2017) which negatively affected the environment and the health of rescue workers (WHO 2015). Furthermore, the surrounding environment was, in addition to radiation and hot reactor cores, dangerous because of a tsunami that had damaged the building (Nationalencyklopedin 2017). In august of 2015 volatile material in a factory in Tianjin, China, exploded and over one hundred people died (Gustafsson 2015). A probable cause to the accident was when fire-fighters tried to extinguish fire in the factory and unknowingly created a violent reaction between the water and chemicals at the scene which lead to explosions (Petersson 2015) that decimated the area.

To facilitate rescue work during accidents like those mentioned, because industrial accidents have a high probability of happening in the future, technology is needed that can be used in environments affected by for instance explosions, fire or chemical spill. Environments that for humans might be highly dangerous and/or relatively inaccessible. Today, drones (unmanned aerial vehicle, UAV) are in combination with different systems used in rescue operations during large scale accidents where they distribute medicine and food to those in need, during military operations like spying or assault (Motlag, Taleb and Arouk 2016) and additionally to give an overview of forest fires (Tomic et al. 2012). The application of drones provides essential support during operations that involves surveillance or needs assessment of a situation (ibid.).

For the drone to navigate, the technology must support both outdoor and indoor navigation as to not reduce the scope of its use. GPS (Global Positioning System) is a system that is generally used in open outside environments to determine the geographical position of senders and has different applications, ex. in vehicles on land, in air or on water (Huh, Shim and Kim 2013). However, GPS used in a closed off environment is problematic because objects block or interfere with the necessary satellite transmission and the GPS-sender loses its connection (NCO-PNT 2017). Thus, navigational system based on GPS is in this situation unreliable.

To be able to navigate and operate a drone where additional infrastructure for communication and navigation probably is non-existent (Tomic et al. 2012) an alternative system for positioning and navigation must be used. For this to be achievable the determination of the positioning of a drone could be based on sensor fusion which is to integrate and use different kinds of sensors together – cameras, ultra sound and laser scanner, for navigation (Hightower and Borriello 2001). These when applied to a drone can be used to scan and map the surrounding environment. Based on the sensor input a map of the unknown environment could be created while simultaneously being used as a navigational tool for the drone to navigate its surroundings.

1.2 Problem

Amid catastrophes and accidents of different kinds rescue operations take place in inhospitable and dangerous environments. Radiation, fire, chemicals and risk of collapse or drowning are some of the dangers that, depending on the accident, rescue workers can be exposed to. Lack of knowledge of the environment and the site of the accident increases risk of exposure to previously mentioned dangers. Something, for example a drone with associated system, is needed to map and collect information about the surroundings to facilitate the rescue operation and reduce the risk of harm. Since accidents happen in buildings or demand that rescue operations must be performed inside it is problematic to use GPS to decide the location of a drone. Therefore, an alternative method for navigation and localisation must be used, were a drone in some way with the help of sensor input can map and then navigate in an unknown environment. The problem that must be solved for this to succeed is called *Simultaneous Localization and Mapping* (*SLAM*) (Dissanayake et al. 2001).

1.3 Research question

How can automated drones be used with a purpose to map and graphically present, to humans, inaccessible and/or dangerous environments?

1.4 Aim

The aim of this thesis is to, with limited resources available, develop a system which will be able to map a surrounding area and graphically present the result digitally. The mapping should be performed without any previous knowledge of the area and consideration must be taken to the fact that signals will in some way be blocked by obstacles which makes the use of GPS for localisation unreliable. Furthermore, the system should register and differentiate between materials as to provide knowledge of possible harmful and dangerous objects and their location.

1.5 Purpose

The purpose of this thesis is to develop and explore a system, a drone equipped with a laser scanner in combination with a SLAM-algorithm, to see how useful it is to facilitate and reduce risks during rescue operations. The subject of the thesis is within the area of immersive networking at the institution of Data- and System Sciences at Stockholm's university and will hopefully work as a foundation for further research.

1.6 Constraints

One part of the constraints of this thesis is dependent on the choice of SLAM-algorithm while others touches on the scope of the thesis and the simulation tool. The set constraints are therefore:

- To simulate the usage of a drone with associated equipment.
- To only base the artefact on already existing project with associated code.
- To write code only in Java and Lua.
- To only present the version of SLAM that is used in the artefact.

2 Extended background

This chapter aims to provide an understanding to the scientific foundation and the specific research area, methods for localisation and position determination and more specifically the solution to the *Simultaneous Localization and Mapping*-problem, this thesis is built upon. Some previous research and development within the SLAM-area will also be presented.

2.1 Scientific foundation

2.1.1 Localisation methods

There exist three principal theoretical methods for localisation – to measure the closeness relative a unit, scene analysis and triangulation. Localisation systems can use these separately or in conjunction with each other to increase the accuracy. (Hightower and Borriello 2001) They will be presented below to get a better understanding of how they can build a foundation to a system that will replace GPS.

2.1.1.1 Closeness

This method is based upon that a physical unit in some way can sense when an object comes within its reach (ibid.). Examples of this are touch- or press sensors that can sense when an object comes into direct contact and Apple’s iBeacon, which senses when an iPhone with a certain application is nearby (Apple, Inc. 2017).

2.1.1.2 Scene analysis

From an object’s observation, by for example camera or radar system, of scenes of its environment it’s possible to analyse and draw conclusions where the object or items nearby are located. These observations are simplified so that characteristics, for instance silhouettes of a mountain, are visible and used for comparison to calculate the location. (Hightower and Borriello 2001)

There are two different kinds of scene analysis – differential and static. A differential scene analysis is that the observer capture scenes at the same time as it moves. A comparison of the different scenes is then made, to determine what separates them. These differences represent the movement pattern of the observer and its position is calculated in relation to the characteristics of the scenes. A static scene analysis is, in contrast, that the observed characteristics are compared to an already pre-defined data set that contains of objects and their respective position to map them to the observed ones. (ibid.)

2.1.1.3 Triangulation

This method uses the geometric properties of triangles to calculate an objects position. This can be made two different ways – *angulation* and *lateration*. For angulation, the calculation is based on two angles relative a reference vector and the distance between two reference points, see Figure 3.

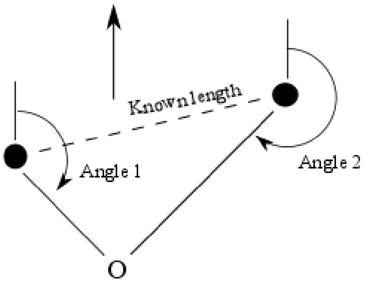


Figure 3: This figure illustrates angulation where angle 1 and 2, relative to a reference vector and the distance between two points, are used to localise object "O" (Hightower and Borriello 2001).

Lateration uses the distance between an object and three reference points, see Figure 4, to calculate the position of an object. In practise, there are three general ways this can be done – to physically measure the distance between an object and several points, to measure the time it takes for an object (ex. ultra-sound) with known velocity to travel to a specific point and to measure the weakening of the intensity of a signal (ex. radio waves) at a specific point. (Hightower and Borriello 2001)

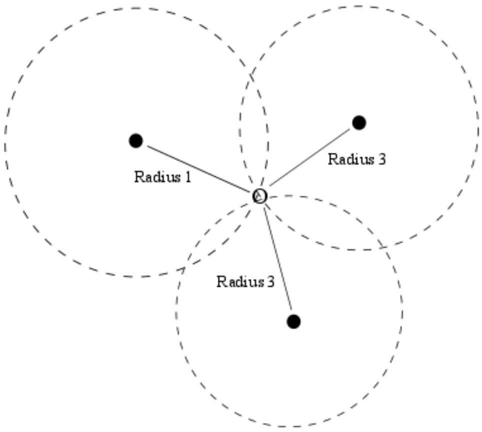


Figure 4: To calculate a position in two dimensions with the help of lateration the distances between object "O" and three reference points are needed (Hightower and Borriello 2001).

2.1.2 The SLAM-problem

As established in section 1.1, GPS is an inadequate localisation method during navigation indoors. Therefore, a drone equipped with a laser scanner will be used to capitalise on a solution to the SLAM-problem as an alternative localisation- and navigational method.

Simultaneous localisation and mapping (SLAM) is, per the authors Durrant-Whyte and Bailey (2006, p. 100) "a process by which a mobile robot can build a map of an environment and at the same time use this map to deduce its location". The problem is here presented, with regards to the work made by Dissanayake et al. (2001) and Durrant-Whyte and Bailey (2006), to give the reader a theoretical understanding of it.

- k – time
- x_k – state vector that describes the position and the orientation of the vehicle
- u_k – control vector that is applied to time k-1 to set the vehicle in the state of x_k at the time of k
- m_i – vector that describes the position of number i object
- z_{ki} - an observation that the vehicle takes of the position of number i object a time k

- $X_{0:k} - \{x_0, x_1, \dots, x_k\} = \{X_{0:k-1}, x_k\}$: the positional history of the vehicle
- $U_{0:k} - \{u_0, u_1, \dots, u_k\} = \{U_{0:k-1}, u_k\}$: the history of the control input
- $m - \{m_1, m_2, \dots, m_n\}$: collection of every object
- $Z_{0:k} - \{z_1, z_2, \dots, z_k\} = \{Z_{0:k}, z_k\}$: collection of every observation of objects

The SLAM-problem demands that the probability distribution

$$P(x_k, m | Z_{0k}, U_{0k}, x_0) \tag{1}$$

is calculated for every *k*. Since both the position of the vehicle and the object is known, the observational model describes the probability to make an z_k observation. This is shown by

$$(x_k | x_{k-1}, u_k). \tag{2}$$

The movement model of the vehicle can be described in the terms of a distribution of the probability of its state transitioning, which is shown by

$$P(x_k | x_{k-1}, u_k). \tag{3}$$

The implementation of a SLAM-algorithm is made through a standard two-step recursive prediction correction:

2.1.2.1 Update of time

$$P(X_k | Z_{0:k-1}, U_{0:k}, x_0) = \int P(X_k | X_{k-1}, u_k) \times P\left(X_{k-1,} m \middle| Z_{0:k-1}, U_{0:k-1}, x_0\right) dx_{k-1} \tag{4}$$

2.1.2.2 Update of measurement

$$P\left(X_{k,} m \middle| Z_{0:k}, U_{0:k}, x_0\right) = \frac{P(z_k | X_k, m) P(X_k, m | Z_{0:k-1}, U_{0:k}, x_0)}{P(z_k | Z_{0:k-1}, U_{0:k})} \tag{5}$$

2.1.2.3 Conclusion

To, on a theoretical level, solve the SLAM-problem a representation of both the observational model (2) and the movement model (3) must be found, which allows consistent and effective calculation of the distributions at (4) and (5) (Durrant-Whyte and Bailey 2006). More information about different solutions and previous research are presented in section 2.2.

2.2 Previous research

Ever since the explication of the SLAM-problem during the 1980s, where methods for probability first were introduced within the areas of artificial intelligence and robotics, has the subject been researched and studied. Several different solutions depending on the situation have been developed. (Durrant-Whyte and Bailey 2006) Two of the most important solution methods (ibid.) to the SLAM-problem are EKF-SLAM (Dissanayake et al. 2001) and FastSLAM (Montemerlo et al. 2002). EKF-SLAM is based upon that representation is in the form of a state-room-model with added Gaussian-noise which leads to the usage of an extended Kalman filter, hence the name EKF-SLAM. The representation of FastSLAM, on the other hand, is used to describe the movement model of the vehicle as a more general Gaussian probability distribution. This leads to a Rao-Blackwellized particle filer is used. (Durrant-Whyte and Bailey 2006)

SLAM has been implemented in everything from underwater and airborne systems to robots that travels both indoors to outdoors (ibid.). One example of an airborne implementation is the work of Grzonka, Grisetti and Burgard (2011) where a small quadrotor system by itself, with the help of a laser scanner and a Kalman filter-based SLAM-algorithm, can navigate indoors. In comparison, one example of a more lightweight algorithm with less calculational cost is the one developed by Nguyen et al. (2006) which is attuned for indoor use and tested with land born robots. Huh, Shim and Kim (2013) has made an integrated system consisting of a camera and a laser scanner fitted on a drone which makes the drone able to, with the help of a SLAM-algorithm, move and navigate in real time.

2.3 Conclusion

Previous research has consisted of developing new algorithms based on previous research for them to then be used as navigational support for both land- and air born robots/drones. Moreover, drones have together with different detectors been developed for outdoor use. This thesis wants to combine the mapping- and imaging properties of a SLAM-algorithm together with a drone equipped with appropriate sensors to both map and create an image of the surrounding area. If possible, the system should also sense what kind of objects and materials exist, the difference between them and where they are in relation to the drone. Through the development and evaluation of an artefact this thesis will show the possibilities that with relatively limited resources exist to create a result that hopefully answers the research question in a satisfying manner.

3 Methodology

This chapter addresses which research method, *design science*, that was deemed most appropriate (and why) when trying to solve the explicated problem that was presented in section 1.3. A thorough description of design science, the design science activities and to the study relevant ethical issues are presented.

3.1 Choice of method

To adequately present a solution to the stated research question, an artefact must be developed. This artefact will be based on previous research and existing technology with the aim of solving a practical problem in a real-world situation. A research method that is developed for this purpose is design science.

Compared to empirical science, where researchers try to describe, explain and predict the world, design science involves designing and developing artefacts to improve already existing practice or to solve practical problems (Johannesson and Perjons 2014). Since the research question of this study requires that a technical solution is developed, design science was considered a reasonable research method and was therefore chosen to be the main strategy of this study.

Due to several limitations, including time and economy, the developed artefact will not result in a physical product but a simulated environment. Simulation is a research strategy that, unlike empirical research strategies, does not aim to study a natural reality, but an imitation. Compared to experiments, which investigates the relation between cause and effect, a simulation does not require a statement that is subject to the investigation but rather an analysis of a complex system is conducted (ibid.). Since computers and code for an appropriate SLAM-algorithm was available, the simulation is performed in a computer environment to imitate the behaviour of a real system.

Another possible candidate in the choice of method is action research, since the study aims to solve a practical problem, which characterizes the mentioned method. Action research is often conducted locally inside an organisation or business to increase the researchers understanding of the operation. Common goals of action research are for example to increase efficiency of routines, improve results or increase the safety of the business (ibid.). Apart from that, action research often requires more time and resources than available for this study. Hence, conducting an action research was not considered feasible.

3.2 Design science process

The design science framework consists of five activities – explicate problem, define requirements, artefact design and development, demonstrate and evaluate the artefact. These activities are presented in comply with Johannesson and Perjons (2014). Worth noting is that they are possible to perform both sequentially and as an iterative process (ibid.), which was done here.

3.2.1 Explicate problem

This activity consists of investigation into and analysis of a practical problem which must be presented and motivated to show how significant it is in a local and global context. The underlying causes should also be identified and analysed to improve the solution (Johannesson and Perjons 2014).

A problem can be explicated and motivated based on collected data from multiple sources – through interviews or surveys with stakeholders affected by the problem, or document studies. Interviews and surveys often require more resources (especially time), since stakeholders must be identified and contacted to gather their views and thoughts on the problem. (ibid.) Therefore, document studies of previous research on the subject was considered most suitable for this study, and chosen as the data collection method for this activity.

3.2.2 Outline artefact and define requirements

The goal of this activity is to discover an appropriate solution to the explicated problem as to identify and outline an artefact. After outlining, requirements required for the artefact to solve are defined. A requirement is a characteristic, functional or non-functional, of the artefact that will be of importance during the design and development. Requirements depend on what kind of problem is explicated, what technology is available, previous research or the stakeholders´ perception of the problem. (Johannesson and Perjons 2014)

The requirements can be collected through interviews or surveys with suitable stakeholders although with same drawbacks as in section 3.2.1. To assign context to the artefact that supports its originality and significance it’s important to base requirements on previous research collected through document studies. (ibid.) Thus, document studies will be used as data collecting method during the requirements gathering activity for this study since access to stakeholders is limited.

3.2.3 Design and develop artefact

This activity involves creating the artefact that was outlined in previous activity. It should address the problem and fulfil the requirements. It’s important to reflect and motivate choices on design to present the artefact as well done as possible. (Johannesson and Perjons 2014)

The resources used during this activity are in this study based on knowledge with origin in research literature which were other implementations of SLAM-algorithms. During the development of an artefact, different research strategies and methods can be depending on the situation (ibid.). This study is using simulation since the artefact is partly made by a simulation tool.

3.2.4 Demonstrate artefact

The goal of this activity is to investigate how the developed artefact’s ability to solve the explicated problem. This is done by demonstrating the artefact in a real or simulated situation. (Johannesson and Perjons 2014)

A case must be chosen and either be developed by the researchers, be a well-documented case from literature, a real case or a combination. To perform demonstration in a real-world situation is favourable as it allows observing the artefact in its environment, but a created case allows the artefact to be tested in more extreme situations that may be hard to recreate in the real world. Documentation of the demonstration is saved for evaluation and further analysis. (ibid.)

The research strategy chosen depends on the situation. Both action research and case studies could be suitable for this study, since it aims to solve a practical problem in a real situation. Since the creation of the artefact in this study is done in a computer system, simulation is the appropriate research strategy for the demonstration of the artefact. Thus, this activity will take form of simulation of three different scenarios where a drone navigates through an environment which will serve as basis for evaluation.

3.2.5 Evaluate artefact

The goal is to show how well the artefact fulfils the requirements and to which extent it can solve the problem. Any potential side effects should be investigated to ensure no unintentional or harmful impact while using the artefact and possible improvements should be reviewed for future research. (Johannesson and Perjons 2014)

Research strategies applied in this activity can be split into two categories – *ex ante* and *ex post evaluations*. Ex ante-evaluation is to evaluate a non-finalized artefact. Ex post-evaluation requires that the artefact has been finalized and tested. The evaluations can vary from the use of surveys where feedback is gathered from experts, to the use of controlled experiments or observations depending on the situation. When evaluating the artefact, it is very important to take the reliability of the data produced by the demonstration in mind, and to ensure that the data is reproducible. (ibid.)

The evaluation in this study will be done ex post, since an artefact will be present. Results from three simulated scenarios will be evaluated through observation. This sampling is of the explorative kind, which means that the result in each scenario is reviewed and analysed further than for a representative sampling (Denscombe 2010).

3.3 Data collection method

Data can be sorted in two different categories – qualitative and quantitative (Johannesson and Perjons 2014). This study will focus on collection of qualitative data in the form of both document studies and data produced during the demonstration of the artefact. Document studies of previous research have been performed throughout the process and works as a foundation for analysing and discussing the subject at hand.

During this step, it is important to keep in mind the validity of the data – is it suitable to refer to and use in the thesis (with regards to the document studies) and is the result produced by the demonstration of the artefact relevant to the research question.

3.4 Method application

This section describes the application of design science and how its' different parts have been designed to suit the aim and purpose of this thesis.

3.4.1 Explicate problem

It wasn't possible, because of limited time, to investigate and obtain problems from a real-world situation. Instead document studies within relevant areas, rescue operations during industrial accidents, were performed which led to the realisation of the problems surrounding GPS that in turn led to more studies about alternative localisation methods and in the end, the SLAM-problem.

3.4.2 Outline artefact and define requirements

Since a drone equipped with a sensor is the main part of this thesis and due to limited resources (mainly financial) a substitute to a real one had to be chosen. The simulation tool V-REP[1] (Virtual Robot Experimentation Platform) was picked because its compatibility with the OS Windows and is free to

[1] *http://coppeliarobotics.com/*

download. This led to the definition of requirements, functional and non-functional, regarding the behaviour of the both the drone, the laser scanner (in the simulation) and the tool itself. Requirements on the chosen SLAM-algorithm, which led choosing BreezySlam, were also defined. This was a continuous process made not only in the beginning but during the development process when problems about the tool and the algorithm were found.

3.4.3 Design and develop artefact

The artefact consists of two parts – one that maps, the simulation, and one that does the imaging, the algorithm BreezySlam. The creation of simulated environments, the behaviour of both the drone and the chosen laser scanner was made in V-REP in accordance to formulated requirements. During this step a lot of problems with BreezySlam and some parts of the simulation environment, the drone specifically, were detected which led to time consuming efforts to solve them. Some changes to BreezySlam had to be made as to better suit this situation.

3.4.4 Demonstrate artefact

The demonstration of the artefact was made by, in V-REP, running three different simulations where a drone together with a laser scanner mapped the area. The data produced by the laser was used as input in BreezySlam which in turn thus produced three different images.

3.4.5 Evaluate artefact

The process of evaluating the artefact consisted of observing and comparing the images produced by BreezySlam with the in V-REP mapped environment to deduce the grade of precision and analyse eventual faults.

3.5 Ethical aspects

When a research process is performed, there are per Vetenskapsrådet (2002) requirements that should be followed. Individuals participating has the right to be protected from unauthorised eyes and to not be exposed to physical or psychological harm. The purpose of principles of ethics in scientific research is to provide guidelines in how the relationship between researchers and participants should be so, at a conflict, a good balance between the requirements can be determined. The principles consist of, as specified by Vetenskapsrådet (2002), four main requirements:

- *The requirement of information* – The researcher should inform those of the study concerned people of the purpose of the study.
- *The requirement of consent* – Participants of a study have the right to choose over their own participation.
- *The requirement of confidentiality* – Information about people who in some way are a part of the study should be treated with as high confidentiality as possible and personal data should be stored so that unauthorised individuals can't get hold of them.
- *The requirement of usage* – Information collected about individuals during the study is only allowed to be used for research.

Since this thesis doesn't have participants nor deal with their personal information no requirements are broken. Something that is relevant for the thesis is that, since already made code is being used, it's important that the code is taken from an Open Source platform or that authorisation to use it is given by the author(s), as to not break 2 chap. 26 g § in Upphovsrättslagen (1960:729). The code that is used, BreezySlam, is free software that can be modified and used per GNU Lesser General Public License.

4 Design- and development process

This chapter presents the design process – from the explication of the problem, defining of requirements to the development of the artefact.

4.1 Explicated problem

From document studies of industrial accidents, the problem of using GPS was explicated and the conclusion that an alternative localisation tool must be used was draw, see chapter 1. Furthermore, the drone should be able to map the environment as to give necessary information to the people that deals with the rescue operation. This led to further literature studies within the subject of localisation- and navigational techniques and previous research about SLAM.

4.2 Outlined artefact and defined requirements

The definition of requirements has in this thesis been based on document studies about previous research on the SLAM-algorithm and assumptions made in regards to the characteristics of the artefact. The process began at the beginning of the project and continued during so the requirements has in an iterative way been changed or added. This was made because new requirements had to be added when problems during the development were encountered.

The requirements are mainly functional in nature and touches upon essential functionality and possibilities of the simulation tool, the drone and laser scanner and the BreezySlam algorithm.

4.2.1 Simulation tool

The simulation tool is the program in where a drone equipped with a laser scanner and its behaviour is simulated to produce data that is processed by the SLAM-algorithm. To find for this project a relevant, reasonable simulation tool several requirements were defined during the early stage of this phase.

- Compatible with the Window OS.
- Should be able to export laser data.
- Large toolbox
- Free to use

These requirements led to the discarding of several found alternatives. It was common, for instance, that a system was only compatible with a Linux-based OS, as the robot simulation programs Gazebo[2] and ROS[3]. Therefore, V-REP[4] was chosen as a simulation tool.

During the development process, more requirements were formulated due to the discovery of several problems in regards to the functionality of BreezySlam.

[2] *http://gazebosim.org/*

[3] *http://ros.org/*

[4] *http://www.coppeliarobotics.com/*

- The drone should be able to move backwards and forwards.
- The drone should be able to turn.
- The range of the laser scanner should be able to be modified.

A non-functional requirement had to be formulated:

- The range of the laser scanner must be 5 meters.

4.2.2 SLAM-algorithm

These requirements were formulated in the beginning of the process. The algorithm should be:

- Compatible with the Windows OS.
- Able to work with by the simulation tool produced data.
- Be Open Source.
- Able to show a graphical imaging of the mapped area.
- Able to differentiate different objects in the area.

These requirements led to many algorithms that were found during document studies and internet searching were discarded since they were compatible with a Linux-based OS and the authors of this thesis, because of limited resources, only had access to a system with the Windows OS. This together with the fact that the authors were, and are, familiar with the programming language Java, led to the choice of BreezySlam.

During the development process several problems with the artefact were discovered which led to the formulation of additional requirements that are specific to BreezySlam.

- The algorithm should be able to process x- and y-coordinates.
- The algorithm should have a class that represents the used laser scanner.
- The image produced should have a relatively high standard.

4.3 Design and development of artefact

The final version of the artefact consists of two parts – a simulation environment (three different scenarios) constructed in V-REP, and a SLAM-algorithm called BreezySlam. These are presented thoroughly below. To get even more information, visit the GitHub[5] of the thesis.

4.3.1 Simulation environment

V-REP was chosen as tool to simulate three scenarios for the drone equipped with a laser scanner to navigate through and map. The program provides numerous different objects and models for use in the simulation, and gives user control of their behaviour through scripts. The specific model of drone that was chosen was *Quadricopter* equipped with a laser scanner of the type *Hokuyo URG04LX UG01-Fast*, with a scanning area of 240°, see Figure 5.

Figure 5: A UAV equipped with a Hokuyo URG04LX UG01-Fast lidar.

[5] *https://github.com/emsand04/BachelorSU*

The environment that the drone navigates through is built with different objects, mainly *wall section 100cm*, see Figure 6.

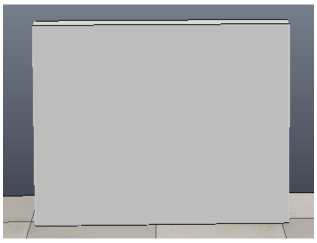


Figure 6: An example of a wall object.

To control the movement of the drone during simulation, a *path* object was used, which the drone followed at the altitude of 50 cm. See Figure 7 for an illustration of a path object.

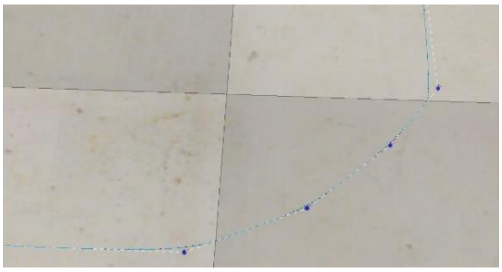


Figure 7: An example of a path object with the shape of a curve.

The velocity and direction of the drone was controlled by Lua-code in a threaded child-script, see Listing 1 for the code controlling the drone in scenario 1. The difference in the code between the different scenarios are the points where the drone turns and which angle of rotation is applied at the points. V-REP has many API functions[6]. The choice to use the function *simRMLPos()* instead of *simFollowPath()* was based on the bigger freedom regarding configurations and movement of the drone, which in turn made keeping the laser facing forward always possible. This was not possible with *simFollowPath()*.

[6] *http://www.coppeliarobotics.com/helpFiles/en/apiFunctionListCategory.htm*

Listing 1: Lua-code that controls the behaviour of the drone in scenario 1.

```
threadFunction=function()
while simGetSimulationState()~=sim_simulation_advancing_abouttostop do
l=simGetPathLength(pathHandle)
posVelAccel={0,0,0}
targetPosVel={l,0}
maxVel=0.2
maxAccel=0.1
maxJerk=50.0
res=0
while res==0 do
    simPosition=simRMLPos(1,0.0001 ,-1,posVelAccel,{maxVel,maxAccel,maxJerk},{1},targetPosVel)
    res,posVelAccel,sync=simRMLStep(simPosition,simGetSimulationTimeStep())
    if (res>=0) then
        dist=posVelAccel[1]
        p=simGetPositionOnPath(pathHandle,dist/l)
        o=simGetOrientationOnPath(pathHandle,dist/l)
        t=simGetSimulationTime()
        simSetObjectPosition(objectHandle,-1,p)

        targetVelocity=simGetObjectVelocity(objectHandle)

        pointOnePosition={-4.1448,-2.2937,0.5}
        pointTwoPosition={0.68540,-2.5728,0.5}
        pointThreePosition={0.84079,-0.83127,0.5}
        pointFourPosition={-2.0592,-0.68127,0.5}
        pointFivePosition= {-2.2098,1.8770,0.5}
        pointSixPosition={-3.9348,1.9270,0.5}

        targetPosition=simGetObjectPosition(objectHandle,-1)
        targetPosOne=math.floor(targetPosition[1])
        targetPosTwo=math.floor(targetPosition[2])
            --First
            if targetPosOne==math.floor(pointOnePosition[1]) and target-
PosTwo==math.floor(pointOnePosition[2]) and t > 0 then
                eulerAngle1={-math.pi/2,0,0}
                simSetObjectOrientation(objectHandle,-1,eulerAngle1)
            end
            --Second
            if targetPosOne==math.floor(pointTwoPosition[1]) and target-
PosTwo==math.floor(pointTwoPosition[2]) then
                eulerAngle2={0,0,math.pi/2}
                simSetObjectOrientation(objectHandle,-1,eulerAngle2)
            end
            --Third
            if targetPosOne==math.floor(pointThreePosition[1]) and target-
PosTwo==math.floor(pointThreePosition[2]) then
                eulerAngle3={-math.pi/2,0,math.pi}
                simSetObjectOrientation(objectHandle,-1,eulerAngle3)
            end
            --Fourth
            if targetPosOne==math.floor(pointFourPosition[1]) and target-
PosTwo==math.floor(pointFourPosition[2]) then
                eulerAngle4={0,0,math.pi/2}
                simSetObjectOrientation(objectHandle,-1,eulerAngle4)
            end
                --Fifth
            if targetPosOne==math.floor(pointFivePosition[1]) and target-
PosTwo==math.floor(pointFivePosition[2]) then
                eulerAngle5={-math.pi/2,0,math.pi}
                simSetObjectOrientation(objectHandle,-1,eulerAngle5)
            end
            --Sixth
            if targetPosOne==math.floor(pointSixPosition[1]) and target-
PosTwo==math.floor(pointSixPosition[2]) then
                eulerAngle6={0,0,-math.pi/2}
                simSetObjectOrientation(objectHandle,-1,eulerAngle6)
            end
    end
simSwitchThread()
end
end
end
-- Put some initialization code here:
simSetThreadSwitchTiming(200)
pathHandle=simGetObjectHandle('Path')
objectHandle=simGetObjectHandle('Quadricopter_target')
```

To save the data produced by the laser scanner, code was written in a non-threaded child script that extracted the gathered data, see Listing 2. The data was stored in a text file with the extension .dat since it is the format required by BreezySlam.

Listing 2: Lua-code that deals saving of laser data

```
name = "laserdata.dat"
file = io.open(name,"w")
io.close(file)
.
.
.
.
file = io.open(name,"a")
    for i=0,#measuredData/3-1,1 do
        file:write(measuredData[3*i+1]," ",measuredData[3*i+2]," ")
    end
file:write("\n")
file:close()
```

Since the sensor length (*maxScanDistance)* of the laser scanner had to be changed additional code was written in the non-threaded child script of the laser, see Listing 3.

Listing 3: Lua-code that deals with the length of the laser.

```
if maxScanDistance>4999 then maxScanDistance=5000 end
```

4.3.2 BreezySlam

In this study the algorithm BreezySlam has been used. It's developed by Bajracharya and Levy (2014) and is accessible at GitHub[7]. The algorithm reads data from a file, processes it and the result is a graph that shows an imaging of a scanned area. Figure 8 was produced based on example data. BreezySlam was chosen since it's relatively easy to use, written in several programming languages and is Windows compatible. The chosen language was Java which the authors of this thesis is familiar with.

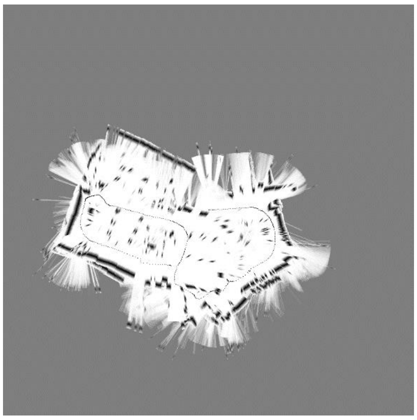


Figure 8: An example of an imaging of a scanned area.

During downloading, installation and testing of BreezySlam several issues were discovered.

- Insufficient documentation.
- The code was not 64-bit compatible.
- Tools for compilation of C++ were missing.
- Some non-usable or erroneous lines of code.

[7] *https://github.com/simondlevy/BreezySLAM*

Since documentation regarding downloading and installation were practically non-existent a lot of time were put towards getting BreezySlam to work. This was made using creative search terms in internet search, many tests and subsequent rewriting of code in both the Make-file and other places, plus the installation of the C++ compiler MinGW[8]. BreezySlam consists of several Java classes where three, Scan, Map and RMHCSLAM, were changed, see Listing 4. The changes made it possible for the algorithm to load certain library files which was a must for it to work.

Listing 4: Example of added Java code made to load library file.

```
static
    {
      try {
        System.load("C:\\Users\\Emma\\Desktop\\BreezySLAM-
master\\java\\edu\\wlu\\cs\\levy\\breezyslam\\algorithms\\libjnibreezyslam_
algorithms.dll");
    } catch (UnsatisfiedLinkError e) {
      System.err.println("Native code library failed to load.\n" + e);
      System.exit(1);
                }
    }
```

In the downloaded package, already written examples with laser data existed. To adapt the code to the laser data created in the simulation environment additional lines of code in the main-class were added, see Listing 5. The laser data is during simulation written to and saved in a .dat-file which the algorithm reads from. The data is in the form of x- and y-coordinates by the laser scanner sensed points. This code makes it so BreezySlam lives up to a set requirement.

Listing 5: Java code written to adapt BreezySlam to this study.

```
//Variables.
private static int SCAN_SIZE = 684;
private static boolean LASER360 = true;
.
.
.
String [] toks = line.split(" +");

if(!LASER360){

long [] odometry = new long [3];
odometry[0] = Long.parseLong(toks[0]);
odometry[1] = Long.parseLong(toks[2]);
odometry[2] = Long.parseLong(toks[3]);
odometries.add(odometry);
               }

double[] floatScan = new double[SCAN_SIZE*2];
int [] scan = new int [SCAN_SIZE];
int [] scanScan = new int[SCAN_SIZE];
ArrayList<Integer> lengthScan = new ArrayList<>();
double length;

if(LASER360){
    for(int a= 0; a<toks.length; a++){
        floatScan[a] = (double)(Float.parseFloat(toks[a]));
    }
//Here the calculation of distance is made, based on xy-coordinates taken from
//file.
//c = sqrt(x^2 + y^2), Pythagorean theorem.
    for(int c=0; c<SCAN_SIZE*2;c=c+2){
        length = Math.floor(Math.sqrt(Math.pow(SCALE*floatScan[c],2) +
Math.pow(SCALE*floatScan[c+1],2)));
        lengthScan.add((int)length);
    }
    for (int i = 0; i<lengthScan.size();i++) {
        scanScan[i] = lengthScan.get(i);
    }
    scans.add(scanScan);
}else{
    for (int k=0; k<SCAN_SIZE; ++k)
    {
        scan[k] = Integer.parseInt(toks[24+k]);
    }
scans.add(scan);
```

---

[8] *http://www.minqw.org/*

The sub-class to Laser (URG04LX) that already existed didn't correlate to the used laser scanner so a new sub-class was created – URG04LX_FAST, se Listing 6.

Listing 6: New Laser class.

```
public class URG04LX_FAST extends Laser {
    /**
    * Builds a URG04LX object.
    * Lidar unit.
    * @param detection_margin         number of rays at edges of scan to ignore
    * @param offset_mm                forward/backward offset of laser motor from robot center
    * @return a new URG04LX_360 object
    *
    */
    public URG04LX_FAST(int detection_margin, double offset_mm)
    {
 //scan_size, scan_rate hz, detection_angle degree, distance_no_detection mm,
//detection_margin , offset m
         super(684 , 10, 240, 5000, detection_margin, offset_mm);
    }
    /**
      * Builds a URG04LX object with zero detection margin, offset mm.
      */
   public URG04LX_FAST()
    {
        this(0, 0);
    }
}
```

Beyond the previously mentioned changes, some additions to the main-class were made to make testing easier. The result was an algorithm adapted after the needs of this study.

5 Implementation

To demonstrate and evaluate the artefact an implementation has been made in the form of three different scenarios created in the simulation program V-REP. The scenarios are presented in more detail in this chapter. Note that for every scenario, the length of the laser scanner is 5 meters and its scope is 240°.

5.1 Scenario 1

The first scenario consists of an environment build by several walls where a drone equipped with a laser maps the area as it moves through it with a velocity of 0.2 m/s, see Figure 9. The path is closed, which means the drone after going a full lap ends up where it started. The drone turns with an angle of 90°, since the laser scanner always must face the motion direction.

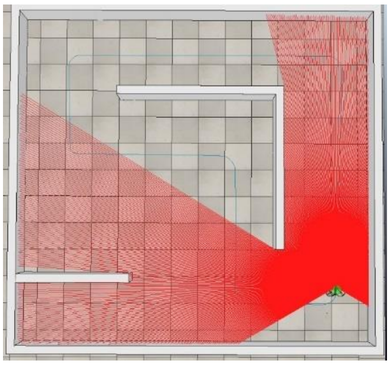


Figure 9: Scenario 1 as seen in V-REP.

To make the drone turn when it should, in the curves, the control positions for the change of the angle and the angles itself are as seen in Listing 7.

Listing 7: The code controlling the drone turning 90° in scenario 1.

```
--The control points that decides when the drone turns.
pointOnePosition={-4.1448,-2.2937,0.5}
pointTwoPosition={0.68540,-2.5728,0.5}
pointThreePosition={0.84079,-0.83127,0.5}
pointFourPosition={-2.0592,-0.68127,0.5}
pointFivePosition= {-2.2098,1.8770,0.5}
pointSixPosition={-3.9348,1.9270,0.5}

--Change of angle.
eulerAngle1={-math.pi/2,0,0}
eulerAngle2={0,0,math.pi/2}
eulerAngle3={-math.pi/2,0,math.pi}
eulerAngle4={0,0,math.pi/2}
eulerAngle5={-math.pi/2,0,math.pi}
eulerAngle6={0,0,-math.pi/2}
```

5.2 Scenario 2

The second scenario is built by similar objects as scenario 1 and the drone follows a closed path which, again, means that after one lap it ends up in the same position as at the start, see Figure 10. In the middle, five objects of the type *cupboard* are standing.

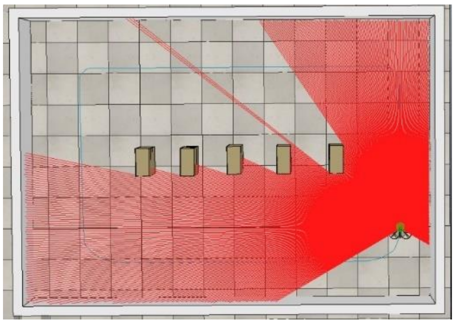


Figure 10: Scenario 2 as seen in V-REP.

To make the drone turn when it should, in the curves, the control positions for the change of the angle and the angles itself are as seen in Listing 8.

Listing 8: The conde controlling the drone turning 90° in scenario 2.
```
--The control points that decides when the drones turn.
pointOnePosition={-4.1448,-2.2937,0.5}
pointTwoPosition={0.68540,-2.5728,0.5}
pointThreePosition={0.81579,0.21873,0.5}
pointFourPosition={-3.7342,0.46873,0.5}

--Change of angle.
eulerAngle1={-math.pi/2,0,0}
eulerAngle2={0,0,math.pi/2}
eulerAngle3={-math.pi/2,0,math.pi}
eulerAngle4={0,0,-math.pi/2}
```

5.3 Scenario 3

The third scenario contains more walls and corridors compared to the other ones and the drone is standing still, see Figure 11. The purpose of this scenario is compare the result of the mapping with a drone that moves and maps at the same time, and to test the artefact's ability deal with the mapping of more objects. Since the drone is standing still, no additional code for motion planning has been written.

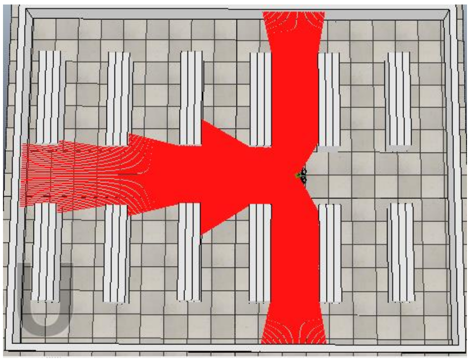


Figure 11: Scenario 3 as seen in V-REP.

6 Result

When the demonstration of the implementation was finished an ex post evaluation was made to determine if the artefact lives up to the set requirements, and if it can reliably answer the research question – *How can automated drones be used with a purpose to map and graphically present, to humans, inaccessible and/or dangerous environments?*

The data collection was made through simulation of the scenarios and the creation of laser data. This data was then run through the algorithm BreezySlam which in turn resulted in different images. These were then trough observation compared to the actual appearance of the simulation environment and is presented in this chapter.

6.1 Simulation

6.1.1 Scenario 1

Figure 12 shows the result of the imaging made by BreezySlam of the data from the mapping of the simulated environment presented in section 5.1.

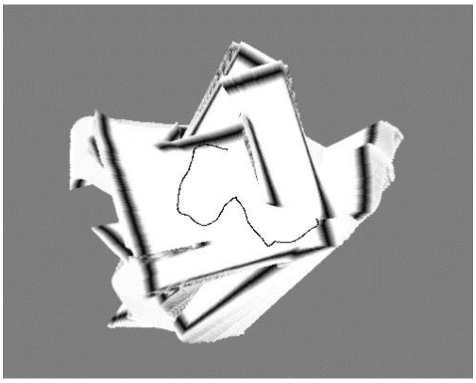


Figure 12: The result of scenario 1.

In Table 1, the parameters that controlled the behaviour of the drone and the laser scanner during the simulation of scenario 1 are presented.

Parameter name	Value
Simulation time	00:1:34:95 (dt=50.0 ms)
Length of the laser scanner	5 meters
Scope of the laser scanner	240°
Scanning frequency	10 Hz
Size of scan	684
Max velocity of drone	0.2 m/s
Max acceleration of drone	0.1 m/s

Table 1: Simulation parameters for scenario 1.

See Listing 9 for an excerpt of laser data that where produced by the simulation and then used by BreezySlam.

Listing 9: Excerpt of laser data from scenario 1.

```
--x- and y-coordinates.
-0.34740483760834 -0.60172235965729
-0.34277963638306 -0.60222315788269
-0.33823502063751 -0.60282111167908
-0.33243280649185 -0.60110187530518
-0.32804608345032 -0.60186898708344
-0.32373008131981 -0.60273069143295
```

Through observation of the imaging of scenario 1, see Figure 13, and the mapping the of simulated environment, see Figure 14, can several differences be seen. Some extra walls have been added, like the one in the lower left corner, and some are missing, like the one at the top. Furthermore, the path of the drone that the algorithm has calculated and imaged as most probable, see the black curvy line in Figure 13 compared to the arrow in the same figure, is also erroneous.

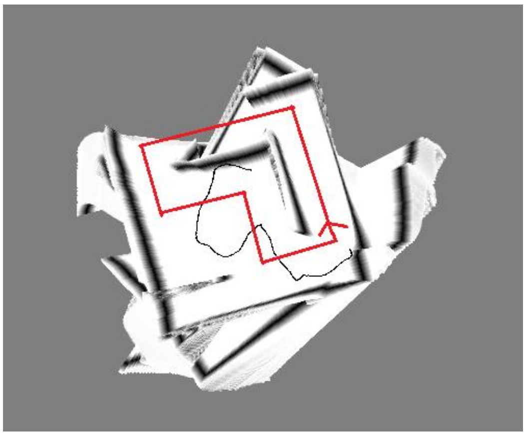


Figure 13: The result of scenario 2 with the estimated way, produced by BreezySlam.

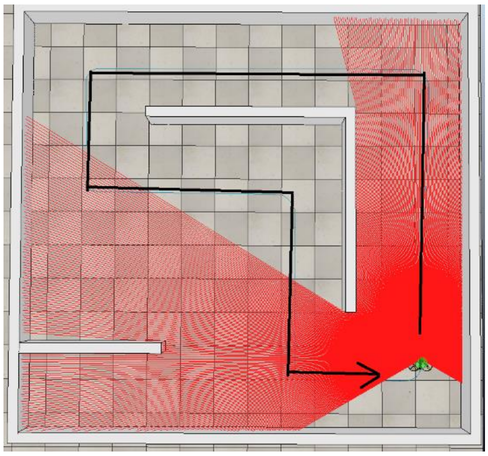


Figure 14: The way of the drone in scenario 1.

6.1.2 Scenario 2

Figure 15 shows the result of the imaging made by BreezySlam of the data from the mapping of the simulated environment presented in section 5.2.

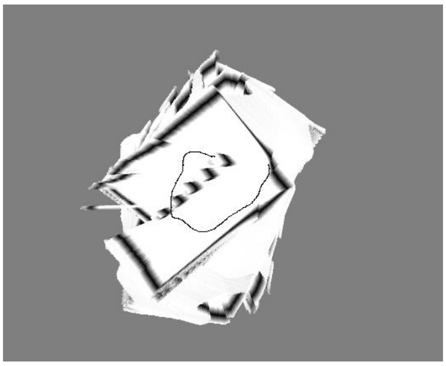


Figure 15: The result of scenario 2.

In Table 2, the parameters that controlled the behaviour of the drone and the laser scanner during the simulation of scenario 2 are presented.

Parameter name	Value
Simulation time	00:1:18:90 (dt=50.0 ms)
Length of the laser scanner	5 meters
Scope of the laser scanner	240°
Scanning frequency	10 Hz
Size of scan	684
Max velocity of drone	0.2 m/s
Max acceleration of drone	0.1 m/s

Table 2: Simulation parameters for scenario 2.

See Listing 10 for an excerpt of laser data that were produced by simulation of scenario 2 and used by BreezySlam.

Listing 10: Excerpt of laser data from scenario 2.

```
--x- and y-coordinates
-0.26085937023163 -0.45182171463966
-0.2573863863945  -0.45219776034355
-0.25397390127182 -0.45264667272568
-0.24961686134338 -0.45135524868965
-0.24632251262665 -0.4519305229187
-0.2430816590786  -0.45257744193077
```

Similar difference as seen in section 6.1.1 can be observed when comparing the result of the imaging of scenario 2, see Figure 16, to the simulated environment, see Figure 17. Some parts of the wall are not drawn properly, and a few non-existing contours have been added. The by the algorithm calculated way of the drone is not correct, as seen when the black curved line is compared with the arrow in Figure 16. All the objects in the middle exist in the image but they are not evenly drawn.

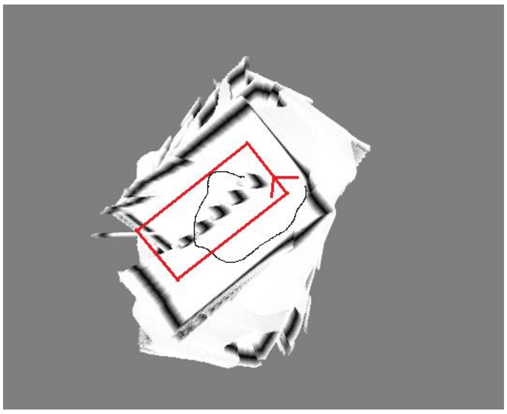


Figure 16: The result of scenario 2 with the estimated way, produced by BreezySlam.

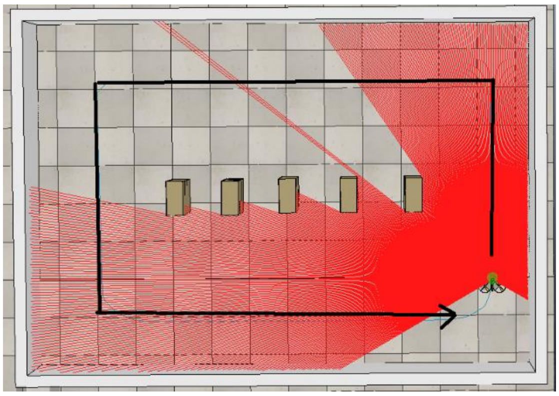


Figure 17: The way of the drone in scenario 2.

6.1.3 Scenario 3

Figure 18 shows the result of the imaging made by BreezySlam of the data from the mapping of the simulated environment presented in section 5.3.

Figure 18: The result of scenario 3.

In Table 3, the parameters that controlled the behaviour of the drone and the laser scanner during the simulation of scenario 3 are presented.

Parameter name	Value
Simulation time	00:0:05:00 (dt=50.0 ms)
Length of the laser scanner	5 meters
Scope of the laser scanner	240°
Scanning frequency	10 Hz
Size of scan	684
Max velocity of drone	0 m/s
Max acceleration of drone	0 m/s

Table 3: Simulation parameters for scenario 3.

See Listing 11 for an excerpt of laser data that was produced in scenario 3 and used by BreezySlam.

Listing 11: Excerpt of laser data from scenario 3.

```
--x- and y-coordinates
-0.404798746l0901 -0.70113170146942
-0.405902713537 22 -0.71312320232391
-0.40724623203278 -0.72581702470779
-0.40369006991386 -0.72994893789291
-0.40539598464966 -0.74378389120102
-0.40737104415894 -0.75845640897751
```

Through observation and comparison of Figure 20 and Figure 19 the conclusion can be drawn that the algorithm in an almost correct way have imaged that which the laser scanner has mapped during the simulation of scenario 3. The objects that are missing are the ones that the laser scanner senses at 0° and 240° respectively.

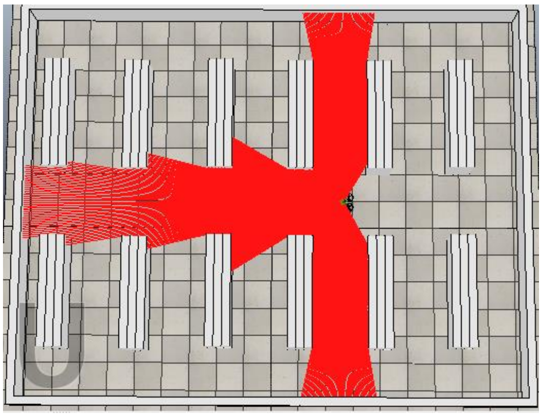


Figure 20: The simulated environment that was mapped in scenario 3.

Figure 19: The result of the imaging of scenario 3.

6.2 Requirements

To investigate whether the artefact fulfils the previously set requirements in section 4.2 they will be evaluated together with the characteristics of the artefact.

6.2.1 V-REP

- Compatible with the Window OS – Yes.
- Should be able to export laser data – Yes, through code in a script it's able to save the data in a file, that can be opened with a text editor and used by BreezySlam.
- Large toolbox – Yes, many different models and objects were available, with several functions that could be used to change the behaviour of an object.
- Free to use – Yes, the specific version used, V-REP PRO EDU V3.4.0 rev1[9], is free to download.
- The drone should be able to move backwards and forwards – Yes, through code and appropriate functions in associated script this was achieved.
- The drone should be able to turn – Yes, but this was achieved in a sub-optimal way, see chapter 5 and 7.
- The range of the laser scanner should be able to be modified – Yes, as can be seen in section 4.3.1.
- The range of the laser scanner must be 5 meters – Yes, this was achieved as seen in section 4.3.1.

6.2.2 BreezySlam

- Compatible with the Windows OS – Yes, after much work.
- Able to work with by the simulation tool produced data – Yes, as seen by the code presented in section 4.3.2.
- Be Open Source -Yes.
- Able to show a graphical imaging of the mapped area – Yes, as seen in section 6.1.

[9] *http://coppeliarobotics.com/downloads.html*

- Able to differentiate different objects in the area – No, since the objects, as seen in section 6.1, are all coloured by the colour black.
- The algorithm should be able to process x- and y-coordinates – Yes, as seen by the code presented in section 4.3.2.
- The algorithm should have a class that represents the used laser scanner – Yes, as seen by the code presented in section 4.3.2.
- The image produced should have a relatively high standard – This is in part fulfilled because the imaging differs slightly to the mapped simulated environment, as seen in section 6.1.1 and 6.1.2, but when the drone is still the standard is higher, see section 6.1.3.

7 Discussion

The aim of this study has been to develop a system that can map an unknown environment and graphically present the imaging. To achieve this a simulation tool and an algorithm, with modifications to suit this study, have been used. The artefact and its result have certain shortcomings in the form of unreliable and inaccurate imaging but forms a foundation for future research and shows that the aim in part can be achieved with limited resources.

The study has been made in accordance to the research strategy design science, which has had its advantages – the technical aspect, to develop and evaluate an artefact, and the ability to work and draw from previously made research was used here. Unfortunately, the strategy has not been used to its full extent because of limited time and resources. For instance, during the requirement phase it wasn't possible to contact and interview people involved with rescue operations or related work. Additionally, one part of the artefact was developed using a simulation tool instead of a real drone which led to limitations regarding the choice of SLAM-algorithm and how advanced it could be. The more advanced algorithms are generally developed on and are compatible with a Linux-based operating system. Design science was a reasonable strategy to choose but better results could have been made given more resources and time.

Based on the result of this study certain things could have been made differently. A computer with a Linux-based OS should have been chosen as development environment which would have led to an increase in choice, both of simulation tool and SLAM-algorithm. This would probably have decreased the time spent on getting the algorithm to work and give readable result, since better documentation regarding for instance ROS-based algorithms exists. Furthermore, a more advanced tool and SLAM-algorithm would probably have led to improved mapping and imaging. Since more time probably would've been available a deeper analysis of the results could have been made, thus an artefact that on a more satisfactory level fulfilled the set requirements.

7.1 The artefact as a solution

The artefact has certain limitations and shortcomings which has been shown through tests and evaluations. However, the artefact has during the development process improved substantially because of increased knowledge and understanding of its faults. How well does it, though, work as a solution to the formulated research question, seen in section 1.3? Because of its limited functionality and somewhat inadequate results, not that well unfortunately. However, it's worth mentioning that the artefact was developed with limited resources and knowledge. Regardless of this, the artefact can still map and graphically present an image of an area, although not with the precision demanded by a real-world rescue operation situation. To develop an artefact with the aim of fully solve the problem and apply in a real-world situation can therefore be possible with enough expertise and resources.

7.2 Significance

This study has, based on previous work within SLAM, an existing algorithm and the simulation tool V-REP, produced an artefact that combines these elements to investigate how a system with a drone able to map an unknown area can work and be made with limited resources. During document studies of previous research similar systems were found, which were similar in nature and/or more advanced. Example of such work, that involves drones and mapping of surrounding area with SLAM-algorithm, are the one made by Huh, Shim and Kim (2013) and the one made by Tomic et al. (2012).

Hopefully, this thesis can lay a foundation for additional work and show the possibilities that exists within this area. Furthermore, that it shows the significance and the support a drone equipped with the right technology can be during rescue operations.

7.3 Limitations

The study has had limitations that have affected the result somewhat. They were mainly created by choices made regarding what to focus on and develop because of limited time and knowledge.

The use of odometry, which BreezySlam can use in its calculations, is something that wasn't applied in this study and probably negatively affected the result. Odometry data, information about the movement of the robot, is often used when implementing a SLAM-algorithm to improve the result and make the calculations more precise. The algorithm can make more qualified calculations of where the robot is since it will know how much the robot has moved and in what direction since the latest scan, as seen in the work by Bajracharya and Levy (2014). An example of the possibilities with odometry is the work by Lucas (2001). It shows how a robot can manoeuvre a path of 7.5 meters and end up almost at the same starting position. To create the algorithm for calculation of odometry of a drone is unfortunately something that the authors of this thesis had limited knowledge of and therefore was not done in this study.

Another limitation has been to only collect data (x- and y-coordinates) through one laser scanner. This led to the artefact not being able to differentiate between materials. With the use of several and different kinds of sensors more information of the environment can be collected and the type of object possibly be determined. For instance, heat sensors can determine if there is fire in the area. Furthermore, the chosen laser scanner together with BreezySlam could only produce an imaging in two dimensions which leads to additional information being lost/not collected.

7.4 Reliability

The reliability of the results has probably been affected by several factors. One factor that has affected the imaging negatively is that the drone doesn't smoothly turn 90°. This made the laser scanner waggle too hard and the subsequent mapping wasn't made correctly. A SLAM-algorithm, as presented in section 2.1.2, draws a point on a position where it has the highest probability to be relative the laser, which made it so, in this case, additional contours were drawn since it was there because of the waggle they existed earlier relative to the laser, see Figure 13 and Figure 16. The turning of the drone is illustrated through Figure 21 to Figure 24 below.

Figure 21: Start of turn.

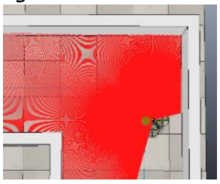


Figure 22: During turn.

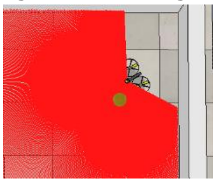


Figure 23: Middle of turn.

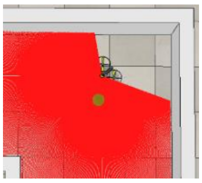
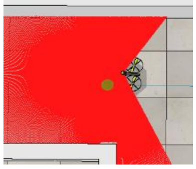


Figure 24: Moving forward.



Since there are no problems with the imaging when the drone stands still, as seen when comparing Figure 20 and Figure 19, underlines the conclusion. But why does the drone have to turn 90° in such a way that it negatively affects the result? The reason is that the laser scanner must always be pointing in the movement direction of the drone, otherwise even more errors occur. This was discovered during the development process.

Another probable contributing factor is that BreezySlam, per Bajracharya and Levy (2014), is developed to use together with a land bound robot, not a flying drone. Code for dealing with the odometry of the robot has been written which affects the imaging in a positive way. In this study, it has not been used since the authors haven't had enough knowledge to adapt BreezySlam to handle odometry of a drone.

7.5 Validity

The aim of this study has been to develop a system that can map and graphically present an area. The requirements have thus been in regards to imaging based on laser data. In this thesis, three scenarios with accompanied result have been made where collection of laser data has been made, which is relevant to the aim and requirements.

The document studies of previous work have been relevant to the subject, since the previous work have been about localisation methods, rescue work and the SLAM-problem etc. The studies have been presented in chapter 1 and chapter 2.

7.6 Ethical aspects

Johannesson and Perjons (2014, p. 190) describes in their book ethical principles and guide lines its recommended for scientific studies to follow. They connect to the ethical requirements presented in section 3.5. To evaluate how this study have followed the recommendations an analysis is presented below.

- *Public interest* – Since this study has involved the creation of an artefact, with the help of a simulation tool and an algorithm, no additional stakeholders have participated. Therefore, no harm that have or will affect stakeholders have happened.
- *Informed consent* – Since no external stakeholders have been asked to participate no consent has had to be asked for.
- *Privacy* – The artefact requires no sensitive information nor any personal details and this study has not dealt with similar information, so privacy of stakeholders has not been affected.
- *Honesty and accuracy* – This study has taken inspiration and information from previously made research and uses an already existing algorithm, which has been presented and accounted for.
- *Property* – The creator of BreezySlam had published the algorithm with the intent of third parties to use and change it. The simulation tool used, V-REP (V-REP PRO EDU V3.4.0 rev1), is free software downloadable from the developer's website.
- *The quality of the artefact* – Since the artefact was developed in a computer environment the risk of physical harm created by the artefact or the study has been low. However, consideration to harmful software or code was made.

8 Conclusion

Based on result from evaluating the artefact can the question, presented in section 1.3, *How can automated drones be used with a purpose to map and graphically present, to humans, inaccessible and/or dangerous environments?* be answered. A drone equipped with laser scanner can, as shown, be used to map areas and present a graphical image, through use of collected laser data and a SLAM-algorithm. It's also possible to make a drone autonomous, but that is not explored here. There are, however, some shortcomings with the developed artefact that makes the use of it in a real-world setting problematic.

The artefact can, in a less reliable was, map and graphically present previously unknown environments. There are flaws in the motion planning of the drone during turning, see Figure 21 to Figure 24, that affects the mapping and imaging in a less satisfactory way than a real-world scenario demands. When comparing Figure 18 to the initial result presented by Bajracharya and Levy (2014), when they also used V-REP, the differences regarding errors are minimal. Nonetheless, the shortcomings of the artefact become more clear when Figure 12 and Figure 15 are compared to the result when Bajracharya and Levy (2014) used real robots, with a laser scanner degree of 360°. For example, the path in Figure 12 is not as exact and the edges in Figure 15 not as sharp. The previously mentioned authors use the robot's odometry in their calculations and the imaging which improves the result, which in this study hasn't been used. Something Bajracharya and Levy (2014) mentioned that they didn't do at the time, to integrate and develop BreezySlam to work with V-REP, is however something this study has accomplished.

The artefact and the algorithm BreezySlam don't differentiate between different types of materials since objects are drawn using the same black colour, see for example Figure 18, compared to for example the algorithm ORB SLAM (Mur-Artal, Montiel and Tardós 2011) where objects are colour coded. The drone is neither equipped with additional sensor that could for instance measure temperature. The laser scanner used only maps the area in two dimensions, compared to the work by Engel, Schöps and Cremers (2014) where the result is a video presentation in three dimensions. The drone is also not autonomous, as the drone in the work of Tomic et al. (2012) is, but follows a created and set path.

The produced results with slightly uncertain reliability have drastically improved during the development process. Unlike previously mentioned studies, such as the one by Tomic et al. (2012) where a real drone was used, have this one been made with limited resources. Despite this have an artefact with the ability to map and graphically present an area been developed, even if on a limited level. Since one part of the aim of this thesis have been just that can that part be seen as fulfilled. Beyond that, possible sources of errors to existing problems have been discovered which means the results through further development could improve.

The conclusion drawn from this study is that the task, to use drones equipped with sensors and a reasonable SLAM-algorithm to map previously unknown areas, is possible. However, the artefact developed in this study is not reliable enough to apply in a real-world rescue operation setting but shows even with limited resources promising results. From the previous research made within this area, a small selection presented in section 2.2, another conclusion is drawn – that the possibility to use more sophisticated methods for mapping and autonomous robots exists.

9 Future research

As previously mentioned this study has been affected by shortcomings that, because of lack of finances and time, were not possible to improve. Because of this, several areas would be appropriate to investigate in the future. Here the areas that haven't been investigated properly will be presented.

The most obvious area for future research is the use of a real drone and laser scanner. This would eliminate all problems that were due to V-REP and take the artefact to a new level. Furthermore, to meet stakeholders such as people working with rescue operations would be desirable since they would provide valuable insight and requirements.

Another area is to improve BreezySlam for flying robots, to make it so that it's able to use odometry of that kind in its calculations.

A third area is to add additional measuring equipment to the drone, since to be able to measure and register different kinds of objects would be beneficent during a rescue operation. This could be, for example, thermal camera, gas sensor or Geiger counter.

The artefact produces an image of the environment after the it has been mapped. This could be improved by showing the imaging at the same time the mapping happens. To implement this other tools are needed, such as OpenCV[10], that in future research could be integrated with the current artefact.

[10] *http://opencv.org/about.html*

Bibliography

Lag (1960:729) om upphovsrätt till litterära och konstnärliga verk. 1960. Sveriges Riksdag.

APPLE, I., Jun 30, 2016-last update, About iBeacon on your iPhone, iPad, and iPod touch [Homepage of Apple, Inc.], [Online]. Available: https://support.apple.com/en-gb/HT202880 [02/24, 2017].

BAJRACHARYA, S. and LEVY, S.D., 2014. BreezySLAM: A Simple, efficient, cross-platform Python package for Simultaneous Localization and Mapping, Washington and Lee University.

CRED, 2017-last update, The International Disaster Database [Homepage of Center for Research on the Epidemiology of Disasters], [Online]. Available: http://www.emdat.be/ [02/04, 2017].

DENSCOMBE, M., 2014. The good research guide: for small-scale social research projects. McGraw-Hill Education (UK).

DISSANAYAKE, M.G., NEWMAN, P., CLARK, S., DURRANT-WHYTE, H.F. and CSORBA, M., 2001. A solution to the simultaneous localization and map building (SLAM) problem. IEEE Transactions on Robotics and Automation, **17**(3), pp. 229-241.

DURRANT-WHYTE, H. and BAILEY, T., 2006. Simultaneous localization and mapping: part I. IEEE robotics & automation magazine, **13**(2), pp. 99-110.

ENGEL, J., SCHÖPS, T. and CREMERS, D., 2014. LSD-SLAM: Large-scale direct monocular SLAM, European Conference on Computer Vision 2014, Springer, pp. 834-849.

GRZONKA, S., GRISETTI, G. and BURGARD, W., 2012. A fully autonomous indoor quadrotor. IEEE Transactions on Robotics, **28**(1), pp. 90-100.

GUSTAFSSON, P., 27 Aug, 2015-last update, Kina trappar upp jakten på de skyldiga för explosionen i Tianjin [Homepage of Sveriges Radio], [Online]. Available: http://sverigesradio.se/sida/artikel.aspx?programid=83&artikel=6241931 [01/27, 2017].

HIGHTOWER, J. and BORRIELLO, G., 2001. Location systems for ubiquitous computing. Computer, **34**(8), pp. 57-66.

HUH, S., SHIM, D.H. and KIM, J., 2013. Integrated navigation system using camera and gimbaled laser scanner for indoor and outdoor autonomous flight of UAVs, Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on 2013, IEEE, pp. 3158-3163.

JOHANNESSON, P. and PERJONS, E., 2014. An introduction to design science. Springer.

LUCAS, G., 2001. Using a PID-based technique for competitive odometry and dead-reckoning. Encoder: Newsletter of the Seattle Robotics Society, .

MICCONI, G., ALEOTTI, J., CASELLI, S., BENASSI, G., ZAMBELLI, N. and ZAPPETTINI, A., 2015. Haptic guided UAV for detection of radiation sources in outdoor environments, Research, Education and Development of Unmanned Aerial Systems (RED-UAS), 2015 Workshop on 2015, IEEE, pp. 265-271.

MONTEMERLO, M., THRUN, S., KOLLER, D. and WEGBREIT, B., 2002. FastSLAM: A factored solution to the simultaneous localization and mapping problem, Aaai/iaai 2002, pp. 593-598.

MOTLAGH, N.H., TALEB, T. and AROUK, O., 2016. Low-Altitude Unmanned Aerial Vehicles-Based Internet of Things Services: Comprehensive Survey and Future Perspectives. IEEE Internet of Things Journal, **3**(6), pp. 899-922.

MUR-ARTAL, R., MONTIEL, J.M.M. and TARDOS, J.D., 2015. ORB-SLAM: a versatile and accurate monocular SLAM system. IEEE Transactions on Robotics, **31**(5), pp. 1147-1163.

NATIONALENCYKLPEDIN, 2017-last update, Fukushimaolyckan [Homepage of Nationalencyklpedin], [Online]. Available: http://www.ne.se/uppslagsverk/encyklopedi/l%C3%A5ng/fukushimaolyckan [01/27, 2017].

NCO-PNT, 10 Feb, 2017-last update, GPS Accuracy [Homepage of National Coordination Office for Space-Based Positioning, Navigation, and Timing], [Online]. Available: http://www.gps.gov/systems/gps/performance/accuracy/ [02/15, 2017].

NGUYEN, V., HARATI, A., MARTINELLI, A., SIEGWART, R. and TOMATIS, N., 2006. Orthogonal SLAM: a step toward lightweight indoor autonomous navigation, Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on 2006, IEEE, pp. 5007-5012.

PETERSSON, T., 2015, 21 Aug. Nästan 200 döda - nu ska olyckan tystas ner. Dagens Nyheter.

TOMIC, T., SCHMID, K., LUTZ, P., DOMEL, A., KASSECKER, M., MAIR, E., GRIXA, I.L., RUESS, F., SUPPA, M. and BURSCHKA, D., 2012. Toward a fully autonomous UAV: Research platform for indoor and outdoor urban search and rescue. IEEE robotics & automation magazine, **19**(3), pp. 46-56.

VAN DIGGELEN, F. and ABRAHAM, C., 2001. Indoor GPS technology. CTIA Wireless-Agenda, Dallas, **89**.

VETENSKAPSRÅDET, 2002. Forskningsetiska principer inom humanistisk-samhällsvetenskaplig forskning.

WHO, 2013. Health risk assessment from the nuclear accident after the 2011 Great East Japan earthquake and tsunami, based on a preliminary dose estimation. World Health Organization.

Stockholms universitet
Institutionen för data- och systemvetenskap
Forum 100
SE-164 40 Kista
Telefon/Phone: 08 – 16 20 00
www.dsv.su.se

Stockholms universitet