HCL Embedded real-time system with applied machine learning

Embedded systems

Aalborg University

Group SW501E18

Title:

Software 5: "HCL Embedded real-time system with applied machine learning"

AALBORG UNIVERSITY
STUDENT REPORT

Group SW501E18

4. September - 2018

Semester:
5. semester
Project theme:
Embedded Systems
Project duration:
04/09/2018 - 20/12/2018
ECTS:
15
Supervisor:
Thomas Bøgholm

Authors:

Casper Weiss Bang

Daniel Moesgaard Andersen

Frederik Spang Thomsen

Nichlas Ørts Lisby

Thomas Højriis Knudsen

Thomas Lundsgaard Hansen

Abstract:
In the Danish educational system, the Arduino platform is often used to introduce students to computer science. The Arduino's hardware is relatively simple and easy to grasp. However, the Arduino language can often be difficult for novices. While the Arduino language is a simplified subset of `C++`, there are still parts of the language that are difficult to comprehend for novices.
This report elaborates and explains the development of the HCL language, a programming language designed to ease the introduction to programming for students and novices, whilst also implementing high-order functionality on the Arduino. HCL does this by emphasizing a resemblance to the English language. A compiler for HCL was developed, as part of the project. The compiler is written in Kotlin, by hand, and compiles to `C++` code.

Number of pages: 14.

Preface

This report is written during the fifth semester of the bachelor's degree in Software at Aalborg University. It is written for the study board of computer science at the School of Information and Communication Technology. The report is written with the guidance of one supervisor.

The topic of the semester is 'Embedded Systems'.

The system to be developed was chosen by the project group.

Terms and abbreviations used in the report:

TM : Turing Machine
NTH : Nice to Have
NXT : Lego Mindstorms NXT 2.0
EV3 : Lego Mindstorms EV3

Contents

1 Introduction **3**

1.0.1 Airports 3

1.0.2 Military 3

1.0.3 Initial Problem 4

2 Analysis **5**

2.1 Embedded platform 5

2.1.1 Lego Mindstorms 5

2.1.2 Single-board computers 6

2.1.3 Takeaways 7

3 Theory **8**

4 Implementation **9**

Bibliography **10**

1 Introduction

Tracking airborne objects can be quite difficult. This difficulty has inspired different sports and hobbies, including skeet shooting. Hitting a target requires the tracking of it, it's velocity and direction, as well as the calculation or prediction of its future trajectory. The trajectory can be impacted by a number of different features, including gravity, wind, initial force, any artificial acceleration. Furthermore, if the object is intelligent in anyway, the path might change procedurally.

Tracking of airborne objects has multiple different applications.

1.0.1 Airports

Estimated thirteen thousand "bird strikes" happen each year, which is the event of a bird colliding with a vehicle, where the majority is aircrafts[1][2]. Even though 65% are estimated to cause little damage, the damages are estimated to cost 1.2 billion dollars each year, world-wide [3]. The majority of these accidents happen at low altitude, during landing or takeoff[3]. Airports utilize different methods of bird control, including killing, trapping, poison and lights[4]. Shooting birds is one of the most common methods, and has been observed to be quite effective, while also reinforcing other methods. This both kills a singular bird, but also scares away other birds. However, it can be expensive in both salaries and effort.

1.0.2 Military

In a military setting hitting a moving airborne target is relevant in a defence context, for instance destroying incoming projectiles or shooting down hostile planes. Compared to the airport-case, the problem can arise at a variety of locations, both on vehicles, boats, planes and tanks, but also on stationary military installations. The American military developed the Phalanx CIWS, Close-in weapon system, which is a radar-guided 20mm Gatling-style rotary cannon, mounted on a base that moves on two axes[5]. These are mounted on ships and can shoot incoming missiles and other airborne objects deemed a threat. The Dutch also developed a similar system called the Goalkeeper CIWS. These are both autonomous.

1.0.3 Initial Problem

Tracking an airborne object has multiple different problem areas that makes it an interesting domain. This makes it interesting to research, and this report will try to look into ways of solving this problem on a smaller scale, in clinical examples, to research the different parameters. Before hitting an airborne target, it is important to estimate its trajectory, and the problem can therefore be simplified to two subproblems.

1. Calculating how many seconds it takes before a projectile can reach the target.
2. Calculating where the target will be in T seconds.

Combined, these two problems means finding the spot where the travel-time of the target meets the travel-time of the projectile.

This report is build on courses in both real-time systems as well as machine learning, and this problem works in the context of both. In regards to machine intelligence, the problem of tracking an airborne target has potential for intelligent analysis by an artificial agent. Likewise, it is a time-critical task to move a weapon so that it aims at the target and hits the target, before the target is out of range or hits the ground.

As hitting the target with a projectile introduces a lot of unknown variables, this report will focus on tracking the target with a laser, as the travel time is near instant (based on the speed of light), and it's less affected by gravity. The report will use more simplistic targets, to remove unpredictable parameters, and to remove the possibility of harming any living creatures.

These considerations creates the following initial problem, whereas the analysis will create a more in-depth set of requirements and a concrete problem statement.

How can a software system track a moving airborne projectile

Analysis 2

2.1 Embedded platform

In the following section will examine the embedded platforms, that will be used in the project. For the 5th semester of the Software education, a Lego Mindstorms kit is supplied to each project group. Therefore the analysis will primarily be based around the Lego Mindstorms platform, with a few additional platforms also considered.

To determine the best platform, the requirements for the project will need to be considered.

1. At the very least, the platform must be able to perform a targeting calculation, based on a trained machine learning model.
2. Be able to interface with motors, most likely Lego.
3. Potentially be able to train a machine learning model, which is used for the prediction on which the targeting is based.

As image recognition is an expensive computation, the current idea is to offload this to an external server or computer, which will compute the current object position and send that to the embedded device.

The current idea is to use the Lego Mindstorms platform for the robotics part of the solution, and use a single-board computer for image object tracking and the machine learning model training.

2.1.1 Lego Mindstorms

Lego Mindstorms is a programmable computer kit, made by Lego. The platform is based around a central control computer, called the Intelligent Brick. It also includes a variety of sensors, motors and connection cables. Each generation further improves on the concept, adding more computational power and better sensors.

The Lego Mindstorm platform, comes with a graphical drag and drop interface. However, various different compilers for many different languages exist. As such the choice between different generations, does not come down to language preferences.

The choices for the 5th semester are the second generation, the Lego Mindstorms NXT 2.0 kit and the third generation, the Lego Mindstorms EV3 kit.

Lego Mindstorms NXT 2.0

The Lego Mindstorms NXT 2.0 (NXT) is the second generation Lego Mindstorms kit. The NXT Intelligent Brick is a micro computer based on the 32-bit ARM7 microprocessor, with 256 kbytes of flash memory and 64 kbytes of RAM[6]. It includes a variety of sensors, such as touch, sound, light and ultrasonic (vision) sensors. On top of this, the consumer kit also includes servo motors and lamps.

A potential major disadvantage of the NXT 2.0, is that the platform is rather old. This means that problems could arise, especially in regards to compatability with newer platforms. This will be further elaborated on in a later section

Lego Mindstorms EV3

The Lego Mindstorms EV3 (EV3) is the third generation of the Lego Mindstorms line. The EV3 mainly improves on the specifications of the second generation, adding a significantly more powerful 300MHz ARM9 processor, running Linux, 16 MB of flash memory, which is extendable with a microSDHC card and 64MB of RAM[7]. On top of this, it also adds remote control and WiFi capabilities, as well as more recently updated compilers.

The primary advantage of the EV3 is that its software is up to date and the power it brings to the table. The power is however also its primary disadvantage. As the goal of this project is to design an embedded real time system, a too powerful computer, could potentially mask some of the challenges that normally come with the development of an embedded system.

2.1.2 Single-board computers

Besides the Lego Mindstorms platform, which is the recommended platform for this semester, there is also the option of using single-board computers, such as the Raspberry Pi or the Arduino. Both of these are more general purpose computers, designed for small electronics projects. Both of the platforms include different models, with some being more powerful than others or having more ports.

Arduino

The Arduino is a single-board computer platform, designed for small electronics projects. The Arduino platform is am eco-system, comprised of different boards, that share the trait of being easily programmable in the included language, a subset of `C++`. Arduino boards typically includes a processor, some RAM and flash memory, as well as programmable GPIO pins[1].

The primary advantage of the Arduino platform is that it is easy to work with, and very modular. A potential disadvantage is that it will not interface well with

[1] General Purpose Input Output

the Lego Mindstorms platform, potentially requiring a custom built solution. This should be doable though.

Raspberry Pi

The Raspberry Pi, the same as the Arduino platform, is a single-board computer ecosystem. Like the Arduino, a Raspberry Pi, typically consists of a processor, RAM, flash memory and GPIO pins. It differs from the Arduino, as it the Raspberry's are typically more powerful, than their Arduino counterparts. In fact, Raspberry Pi's are, for all intents and purposes, just weaker desktop computers.

Raspberry Pi's also typically include some form of network connectivity, whether it is a WiFi chip or a LAN port. It also includes USB ports, which means it can use a USB camera, which could be required for the object tracking.

2.1.3 Takeaways

Both the NXT 2.0 and the EV3 are excellent robotics platform, and the choice between the two generations, will primarily be dependent on the limits that the project group imposes on itself. For the image recognition platform, the Raspberry Pi seems like the best choice, as the Arduino, depending on the specific model, is only as powerful as the NXT or the EV3, rendering the Arduino a bit redundant.

Theory 3

Implementation 4

Bibliography

[1] Wikipedia; Various authors, "Bird strike," https://en.wikipedia.org/wiki/Bird_strike, 2018, [Online; Accessed: 18-09-2018].

[2] US Department of Transportation, "Wildlife strike report 1990-2015," https://wildlife.faa.gov/downloads/Wildlife-Strike-Report-1990-2015.pdf, 2015, [Online; Accessed: 18-09-2018].

[3] John R. Allan, Alex P. Orosz, "The costs of birdstrikes to commercial aviation," http://digitalcommons.unl.edu/cgi/viewcontent.cgi?article=1001&context=birdstrike2001, 2011, [Online; Accessed: 18-09-2018].

[4] Programmadirectie Ontwikkeling Nationale Luchthaven, "Bird control at airports," http://www2.vlieghinder.nl/naslagdocs/CDrom/REGELS_SCHIPHOL/2.3_TNL/4.3.4.1_Bird_control_at_airports.pdf, 1999, [Online; Accessed: 18-09-2018].

[5] Wikipedia; Various authors, "Phalanx ciws," https://en.wikipedia.org/wiki/Phalanx_CIWS, 1999, [Online; Accessed: 18-09-2018].

[6] Lego, "Nxt user guide," https://www.generationrobots.com/media/Lego-Mindstorms-NXT-Education-Kit.pdf, 2018, [Online; Accessed: 2018-16-09].

[7] ——, "Ev3 user guide," https://le-www-live-s.legocdn.com/ev3/userguide/1.4.0/ev3_userguide_enus.pdf, 2018, [Online; Accessed: 2018-16-09].