Predicting Human Movement Patterns in an Office Environment

Fredrik Hagnell
Abstract

This project is built on the idea of predicting future human movement in an area. The algorithm’s predictions are based on previous movements in the area which has to be recorded somehow. For this a device with a motion sensor was setup to monitor the movement in a hallway in an office. This data was then used to test and evaluate the prediction algorithm. To give feedback about the movement and how it is changing to the people working in the office the setup device shows sentences on a monitor which describes the movement.

The project resulted in a fully working application which measures people walking by, both when and how fast, and predicts future movement. Due to time constraints of the project the device was only up and running for two weeks. This is enough time to get some understanding of how well the prediction algorithm works, but a longer experiment time would have further helped the evaluation. The results showed that the algorithm can predict most of the events during the day, but is bad at predicting sudden spikes or other unusual behavior.

Keywords
Predict, human movement, motion sensor
Abstract


Projektet resulterade i en fullt fungerande applikation som mäter folk som går förbi, både när och hur snabbt, och förutser framtida rörelse. På grund av tids begränsningar i projektet så var anordningen bara uppe och mätte data i två veckor. Detta är tillräckligt mycket tid för att få någon förståelse över hur bra förutsägelse algoritmen fungerar, men en längre experiment tid skulle ha hjälpt utvärderingen. Resultaten visade att algoritmen kan förutse de flesta händelserna under dagen, men är dålig på att förutse plötsliga spikar eller annat ovanligt beteende.

Nyckelord
Förutse, mänsklig rörelse, rörelsesensor
Table of Contents

1 Introduction .................................................................................................................. 1
  1.1 Background ............................................................................................................. 2
  1.2 Problem .................................................................................................................. 2
  1.3 Purpose ................................................................................................................... 2
  1.4 Goal ........................................................................................................................ 2
  1.5 Benefits and Ethics ............................................................................................... 2
  1.6 Methodology / Methods ......................................................................................... 3
  1.7 Delimitations ........................................................................................................... 4
  1.8 Outline (Disposition) ............................................................................................. 4

2 Tracking of Movements ............................................................................................... 5
  2.1 Related work ........................................................................................................... 6
      2.1.1 Conclusion ...................................................................................................... 7

3 Methodology ................................................................................................................ 9
  3.1 Research Strategy .................................................................................................. 9
  3.2 Data Collection ...................................................................................................... 9
  3.3 Data Analysis ......................................................................................................... 10
  3.4 Quality Assurance ............................................................................................... 10

4 Development .............................................................................................................. 11
  4.1 Sensor .................................................................................................................... 11
      4.1.1 Passive IR sensor ............................................................................................ 11
      4.1.2 Ultrasonic sensor ........................................................................................... 12
      4.1.3 Camera ............................................................................................................ 14
      4.1.4 Distance measuring sensor ............................................................................. 15
      4.1.5 IR Break-beam sensor ..................................................................................... 16
      4.1.6 Conclusion ...................................................................................................... 17
  4.2 Platform setup ....................................................................................................... 18
      4.2.1 Arduino ............................................................................................................ 18
      4.2.2 FPGA ............................................................................................................... 18
      4.2.3 Raspberry Pi .................................................................................................... 18
      4.2.4 Conclusion ...................................................................................................... 18
  4.3 Programming Language ....................................................................................... 19
  4.4 Selected texts ........................................................................................................ 19
  4.5 Word lists .............................................................................................................. 19
      4.5.1 Movement word list ....................................................................................... 20
      4.5.2 Change of movement word list ....................................................................... 20
  4.6 Graphical feedback ............................................................................................... 21
  4.7 Data collection ....................................................................................................... 22

5 Prediction ..................................................................................................................... 23
  5.1 Data ....................................................................................................................... 23
  5.2 Weights ................................................................................................................. 23
  5.3 Algorithm ................................................................................................................. 24

6 Results ........................................................................................................................ 25
  6.1 Prediction results .................................................................................................. 25
  6.2 Summary ............................................................................................................... 32
7 Conclusions........................................................................................................33
  7.1 Discussion....................................................................................................33
8 Future Work.....................................................................................................34
References .........................................................................................................35
Appendix A ....................................................................................................... 1
Appendix B ....................................................................................................... 1
1 Introduction

This project is built on the idea of measuring the rhythm of human movement in an office area in order to predict future movement. In order to track the movement a motion sensor was setup which detects what time someone walks by and how long they were in the monitored area, i.e. with what speed they were walking.

The program also gives feedback to the people walking by the sensor by showing a sentence describing the recent movement trends. The sentences shown are taken from scripts of famous movies obtained from an online source [1]. The sentences are selected based on the words in them. If a sentence contains the word “slow” for example, it might be a fitting match for a slow movement. For example “It was a slow day”. Words which describe a certain movement are stored in a list with number values defining what movement they fit.

All collected data, timestamps and how long the person was in the sensor’s area, is stored and can be used to predict movement on a future date. The prediction is more heavily influenced by dates which more closely relate to the prediction date. Dates which are on the same weekday will have more influence and the closer a date is to the prediction date the more influence it will have on the result.

The sensor was setup and monitored the area for about two weeks, from Monday the 12th to Friday the 23rd of October 2015. The data collected was used to evaluate the prediction algorithm. The prediction algorithm was evaluated by using parts of the data as input and predicting a date which had been measured. This way the prediction can be compared to the real life results.

Being able to predict future movement can be very useful and has been researched before [2][3]. It can help to find work patterns in an office and with this information help increase productivity. It can also help you find problems in the movement which can lead to better work environment if addressed. This project focuses on the movement prediction in an office, but the same principles can be applied to other situations, such as the public transport system where busses and trains can be sent to stations the prediction algorithm thinks there will be a heavy load. By being able to predict the future you can better plan for it.
1.1 Background
Researchers at Mobile Life Centre are trying to find ways to bridge the gap between human experience and our experience with digital interactions. One of the ways they aim to achieve this goal is to express human movement in a language-based format. Another way is for a program to be able to understand and predict future human movement.

1.2 Problem
The prediction of human movement is something that has been researched before in many different situations. However, the prediction of human movement of non-specific people in a fixed area is a problem that is not well researched. It is possible that there are many inefficiencies that can be improved such as a bad work rhythm in an office. Being able to accurately measure and predict future movements can help show and solve these inefficiencies.

The problem this project aims to solve is how can we accurately predict future movement in an office area?

1.3 Purpose
The purpose of this thesis is to present a way how to accurately measure movement and predict future movements in a fixed area.

1.4 Goal
The goal of the project is to create a device which collects data of the movement of people in a small area and is able to predict future movement. This device will be presented to the researchers at Mobile Life Centre. The device will give Mobile Life Centre a better overview of the work rhythm of its employees.

1.5 Benefits and Ethics
The degree project is made at and for the company Mobile Life Centre so they will benefit from it. Hopefully they will find use for it in the future by further working on the project or by using parts of it in future ideas.

There are a few ethical considerations in the project. Since it will monitor movement in an office it is important that no personal data be saved. The people working in the office were told in advance of the monitoring and what it was supposed to achieve. The dates selected for the result examination should be picked without any bias to avoid only reporting positive results. The prediction algorithm should also
not be altered just to fit the measured data. There can not only be experiments that are designed to make the prediction algorithm look good.

1.6 Methodology
There are two main categories of research methods: quantitative and qualitative. Quantitative methods are for projects which uses numerical data with large datasets. Qualitative methods, on the other hand, use non-numerical data and small data sets. These two categories can also be combined into a mixed approach if it fits the project [4]. While this project does use big datasets of numerical data, the importance lies in understanding the data and evaluation of the prediction algorithm. Therefore, the qualitative method is the most suitable one and will be used.

A philosophical assumption is the starting point for the research [4]. There are three different philosophical assumptions for qualitative research methods: realism, interpretivism and criticalism. Realism assumes that things are known and are not dependent on biases of the observer. Phenomenon provides credible data and facts. Interpretivism assumes that reality is only accessed via social constructions. This assumption is used to discover the meanings people assign to phenomenon. Criticalism focuses on the oppositions, conflicts and contradictions in society with the goal to eliminate their causes. The realism philosophical assumption is the most suitable one for this project and will therefore be used.

Research methods provide procedures for accomplishing research tasks [4]. Since this project uses realism as philosophical assumption there are three different research methods to choose between: Fundamental, applied and non-experimental. Fundamental research, also called basic research, focuses on fundamental principles and testing theories. Applied research involves solving known and practical problems and the results are related to a particular situation. Non-experimental method describes or predicts behavior and draws conclusions for the situation. The applied research method suits this project well and will therefore be used.

Research approaches describe how to draw conclusions and how to establish what is true or false [4]. The two main research approaches are deductive and inductive, but there is also a third approach called abductive which is a mix of the other two. The deductive approach is about deriving conclusions from known premises. The inductive approach is about establishing a general proposition from facts. The mixed approach, abductive, is about deriving likely conclusions from an incomplete set of observations. For this degree project, the abductive approach is most suitable and will therefore be used.
1.7 Delimitations

The best way to evaluate the project’s result would be to have the prototype up and gather data under a long period of time, preferably months. This would also give the algorithm more data to use when predicting. Unfortunately, the prototype will only be able to be up and running and gathering data for about two weeks. This still gives plenty of data for the algorithm to use when predicting and for the evaluation process, but will not lead to as good a result as if it would be up for several months. The reason for this limitation is due to time constraints of the project.

1.8 Outline (Disposition)

The thesis starts in chapter 2 with an overview of the project where the background and related work are discussed. In chapter 3 the methods used for the project are discussed in detail.

In chapter 4, the different ways to construct the prototype is analyzed. First which sensor should be used is discussed, different solutions with their pros and cons and finally a decision and the reasoning behind it. Then the general sensor setup is discussed and what will be used to handle the data with pros and cons of the different setup possibilities and finally a conclusion with an explanation why the selected system was selected. What programming language was selected and why is then discussed. Finally, there are some explanation of how the movement is shown to the users and how the data was collected.

In chapter 5 the prediction algorithm is discussed. How the data is collected, stored and used. In the next chapter, chapter 6, the results of the project are presented with comparisons of the predicted data and the actual measured data. In chapter 7 the conclusions of the project are presented. In chapter 8, the final chapter, ways to expand on the project’s idea and take it further is discussed with examples.
2 Tracking of Movements

This project is only interested in the movement in a certain area, not in tracking the movement of specific people over a wide area. Therefore, all data will be non-personal and collected in a fixed position on a limited population. This differentiates from most related work found.

The selected area to be monitored was a hallway near an entrance of an office. The office is Mobile Life Centre which has about 30 employees [5]. The monitored area is located in a position where most of the employees walk past it when they arrive in the morning, go to lunch, get coffee, go to the bathroom and go home in the afternoon.

In order to track the movement a motion sensor was setup which detects what time someone walks by and how long they were in the monitored area, i.e. with what speed they were walking. All collected data is stored and can be used to predict movements of future dates.

The background of this project is that researchers at Mobile Life Centre are trying to find ways to bridge the gap between human experiences and digital interactions. One way they aim to achieve this is to be able to automatically describe data using easily understood text so that it can easily be understood by humans. Therefore, the device built in this project will also give feedback in text format to the people walking by the sensor’s area.

The feedback from the device is text sentences which describe the recent trends in movement in the area and are shown on a monitor. The text sentences are taken from movie scripts obtained from an online source [1]. The sentences are selected based on the words in them. If a sentence contains the word “slow” for example, it might be a fitting match for describing the recent movements if there hasn’t been much activity in the area recently.

The text feedback is a smaller side part of the project. The main objective is to create an algorithm which can accurately predict the movement in the area.
2.1 Related work

This is not the first research project with the goal to be able to predict human movement. Research projects before this one has used patterns in human movement in order to predict future ones. This section covers two related projects which use different ways to gather data of movement to predict future movement to achieve different goals.

In [2] the researchers try to create a home which works as a rational agent. The agent perceives the state of the home through sensors and use device controllers to act upon the environment. It gathers data of the inhabitants in the home by using sensors and then it analyses this data to predict future behavior. Such as turning lights on or off, make coffee and order food.

They propose a variety of algorithms to predict future events. The algorithm most related to this project is an algorithm they call *Smart Home Inhabitant Prediction (SHIP)*. The SHIP algorithm predicts future events by looking at the most recent sequence of events and matching them with sequences in collected histories. The matches in the queue are evaluated based on the match length and frequency. There is also a decay factor in the algorithm to allow for gradual changes in the patterns over time.

In [3] the researchers try to predict the location of a user’s mobile device. In the paper, they overview a number of different algorithms which they classify into two types of approaches: domain-independent and domain-specific.

Their domain-independent algorithms take results from Markov analysis, i.e. that future states are independent of past history. In the paper they define an order-$k$ Markov predictor algorithm which assumes that the next term of movement only depends of the $k$ most recent terms. This means that only the past $k$ history needs to be stored since anything older is considered obsolete. This saves data storage, but if $k$ is set too low the prediction will be unreliable. They also define another kind of domain-independent algorithm similar to the order-$k$ Markov predictor except that $k$ is a variable allowed to grow to infinity. If $k$ is a variable which does not have a maximum value, no data can be considered obsolete and discarded. It is still based on the assumption that past history does not impact future states and will thus only increase $k$ if necessary.

Their domain-specific algorithms consider the geometry of user motion as well as the semantics in the user’s movement history. These algorithms distinguish between stationary states, if the user resides at a location longer than some threshold time interval, and transitional states, if the user is moving. Two types of movement
patterns are defined. A Movement Circle is a sequence of locations that begin and end at the same location. A Movement Track is a sequence of locations that both begin and end in stationary states but in two different locations. The algorithm builds up a database of both of these movement patterns for each user. When predicting a future location, the algorithm looks at the recent locations and states and tries to find the best match in the database.

2.1.1 Conclusion
In this project the prediction algorithm should be able to predict the movement in an area on a future day by using data of movement from earlier days. This differentiates from the related work, but some parts of the related work can still be used.

The SHIP algorithm [2] looks at the most recent events and tries to match them with recorded events in the user’s history. This seems like a good idea when predicting events in the next few minutes, but might not work accurately when predicting full days. However, the decay factor they use is a good idea and should work well with this project, i.e. that more recent data should have a bigger impact on the prediction than older data.

The order-\(k\) Markov algorithm [3], which assumes that the next term of movement only depends on the \(k\) most recent terms, is similar to the decay idea, but more extreme. It is possible to combine these two ideas for this project by only looking at the \(k\) most recent days when predicting and having the more recent days impact the prediction more than the older days. This should lead to a more accurate prediction assuming that there are patterns and that patterns change over time.

The usage of different states of movement and movement patterns they use in their domain-specific algorithms [3] are interesting. This project does not track specific people and only covers a specific area which means that the movement states and movement patterns they use cannot be used for this project. However, the idea behind different states can still be used by defining different states. A possible state that can be used is the weekday the movement is happening. All seven weekdays would be different states where movement from the same state would have a greater impact on the prediction than movement from the other states. Separating the states completely, i.e. only using data from Wednesdays when predicting a Wednesday, would leave very little data for the prediction algorithm. It would also mean that the prediction algorithm misses certain patterns from the other days which might be important. Since this project will be setup in an office environment there is a clear difference of movement between
workdays and weekends. This can also be set as a state and since the movement is so drastically different, these states should not impact each other, i.e. data from a Saturday should not impact the prediction of a Monday and vice versa.
3 Methodology

This project uses the qualitative method since the importance lies in understanding the data and the results from the prediction algorithm. The philosophical assumption is realism since this project contains observing phenomenon to collect data which is seen as credible and then develop knowledge from the collected data. The applied research method is used since this project is about solving a known and practical problem with the results being related to a particular situation. The research approach used is the abductive approach. The abductive research approach is the most suitable research approach for this project since conclusions will be derived from an incomplete set of observations.

3.1 Research Strategy

The research strategy is the guidelines for carrying out the research [4]. This includes organizing, planning, designing and conducting the research. This project uses qualitative research which commonly uses surveys, case study, action research, exploratory research, grounded theory or ethnography as research strategy. Surveys describes phenomenon which are not directly observed by collecting information from a population by using questionnaires. Case study involves empirical investigation of a phenomenon. Action research is a systematic, cyclic method. It is performed by actions to contribute to practical concerns in a problematic situation. Exploratory research explores the possibility to obtain as many relationships between different variables as possible by using surveys. It rarely provides definite answers. Grounded theory aims to develop a theory that is grounded in data by systematically collecting and analyzing data. Ethnography method seeks to place phenomenon in their social and cultural context.

For this project the grounded theory is most suitable and will therefore be used. It is the most suitable since the conclusions drawn from this project will be grounded in data. There will be collecting and analyzing of data which will lead to conclusions.

3.2 Data Collection

Data collection methods define how data will be collected for the research [4]. The most commonly used methods for qualitative research is questionnaire, case study, observations, interviews, and language and text. Questionnaires collect data from a population through questions. Case study is an in-depth analysis of a small number of participants and is used together with the case study research method. Observations observe behaviors with focus on situations. Interviews give deep understandings of a problem from
the participants’ point of view. *Language and text* is used for interpreting conversations and meanings in texts.

This project will use the *observations* data collection method since the data will be collected by observing the behavior of a population in a fixed area.

### 3.3 Data Analysis

The data analysis method describes how the data collected will be analyzed [4]. It is the process of inspecting, cleaning, transforming and modeling data. The most commonly used data analysis methods for *qualitative* research are *coding*, *analytic induction*, *grounded theory*, *narrative analysis*, *hermeneutic*, and *semiotic*. *Coding* turns *qualitative* data into *quantitative* data by analyzing transcriptions of interviews and observations. Both *analytic induction* and *grounded theory* are iterative methods which alternate between collections and analysis. *Narrative analysis* concerns literary discussion and analysis. *Hermeneutic* and *semiotic* methods are both used for analyzing texts and documents. *Hermeneutic* is used for text while *semiotic* is used for signs and symbols.

The *coding* data analysis method is the most suitable one for this project. The observations will be turned into quantitative data which will then be used by the prediction algorithm.

### 3.4 Quality Assurance

Quality assurance is the validation and verification of the research material [4]. It includes *replicability*, *validity* and *ethics*.

The collected data cannot be replicated, but regardless of the data collected the final results of the prediction algorithm should have similar accuracy compared to different observations. In other words, the data cannot be replicated, but the results can be.

The monitoring device will be tested before use to make sure it works as intended to avoid any wrong measurements.

The collected data will be non-personal. The device monitoring the participants will not be able to tell the difference between the people.
4 Development

The movement has to be measured in some way. Several different ways to measure the movement were analyzed in order to find the best way of doing it. What type of sensor should be used and what should handle the sensor output, i.e. what should the sensor be connected to?

4.1 Sensor

The sensor has to be able to detect people walking by in a hallway. The hallway is not very wide so the sensor's range does not have to be long. The sensor only has to detect people walking by at a certain point in the hallway, i.e. it does not have to detect people in a large area. It does not have to be able to detect who is walking by, only that someone is walking by.

5 different kinds of sensors were analyzed to find the best suited sensor for this project.

4.1.1 Passive IR sensor

A passive infrared (PIR) sensor works by detecting changes in the infrared radiation that is triggered by any warm object, such as a human, entering the area it covers. The passive part of the name means that it is not emitting anything, it only receives. This means that it is very energy efficient for prolonged idle situations. [6].

A big upside to this type of sensor is that it is very simple to set up and use. You just point it in the direction you want to detect motion. It is also cheap and very energy efficient.

A downside is that it cannot differentiate between people walking next to each other. Those people will only count as one movement according to the sensor.
Fig 1. The sensor triggers when a warm object, such as a human, enters or exits the area

4.1.2 Ultrasonic sensor
An ultrasonic sensor works by continuously sending out waves of ultrasonic frequencies and then listening to the return pulse. By doing this it can calculate the distance to the object in front. If the distance is less than the distance to the wall, the sensor knows that there is something in between the wall and the sensor. [6].

This sensor is, like the PIR sensor, also easy to set up and use, but since it is continuously emitting sound waves it is not energy efficient. It also has trouble differentiating between people walking next to each other.
Fig 2. The sensor continuously measures the distance to the object in front. If this distance is shorter than the distance to the wall an object is passing in front of the sensor.
4.1.3 Camera

Using a camera with face or body recognition to see when someone walks by would solve the problem with people walking next to each other, since it would be able to differentiate between them.

This solution is much more complex than any of the other solution and also much more expensive. There is also an ethics part of it. This is the only solution which tracks individual people and would undoubtedly make some people feel uneasy.

*Fig 3. The sensor tracks the objects in its area to find humans walking by.*
4.1.4 Distance measuring sensor

Distance measuring photoelectric sensors emit a laser light that reflects back to the sensor. By measuring the time it takes for the light to be reflected it can calculate how far away the object in front is. [7]. If the distance is less than the distance to the wall, there is something between the wall and the sensor.

This type of sensor shares the problems the ultrasonic sensor has. It is continuously emitting and it has trouble differentiating between people walking next to each other.

*Fig 4. The sensor continuously measures the distance to the object in front. If this distance is shorter than the distance to the wall an object is passing in front of the sensor.*
4.1.5 IR Break-beam sensor

A break-beam sensor works by having a transmitter send out a beam of light and a receiver across the room which is sensitive to the light transmitted. When the beam is broken, i.e. something passes between the two sensors, the receiver will let you know [8]. The light does not have to be IR, but having non-visible light for humans was seen as better since less attention would be given to it.

A big upside to this type of sensor is that it is very simple, either the beam is broken or it is not. It also covers a very small area, which gives better control of where the motion is supposed to be detected.

A big downside however is that it requires two sensors across the room from each other. This makes it much more difficult to set up properly and limits where it can be set up. It will also count people walking next to each other as one person since the beam will only be broken once.

![Diagram of a break-beam sensor](image)

*Fig 5. The sensor triggers when an object blocks the beam.*
4.1.6 Conclusion

Of the different sensors analyzed, the passive infrared sensor was selected as the most suitable for the project. The only downside of it was that it cannot differentiate between people walking next to each other. All the other solutions shared this problem, except for the camera one which was discarded as it was seen to make things unnecessarily complicated. A PIR sensor is very easy to set up, cheap to buy and, due to it being passive, energy efficient. There is also no way to gather personal data from it, which makes it less intrusive to the people it is monitoring.

Another big upside of the PIR sensor was that it can measure how fast the person is walking by measuring the time he or she is in the sensor's area. This creates more opportunities for the movement analysis.
4.2 Platform setup

The sensor needs to be connected to something which can handle the data and convert it into language. For this 3 different setups were analyzed

4.2.1 Arduino

Arduino boards are able to read inputs and send outputs. There are many different kinds of boards. Their hardware and software are made to be easy to use [9]. However, being easy to use comes with its drawbacks. While it is easy to create a simple project with Arduino, it is difficult to create a software intense project. To show an output on a screen or with a projector also comes with its difficulties. Arduino can be connected to a screen or a projector with a VGA-cable, but it takes a lot of tinkering to make it work and have it look good.

4.2.2 FPGA

The Field Programmable Gate Array (FPGA) boards are more complex than the Arduino boards. The FPGA boards also come in many different variants [10].

Because of the FPGA boards' complexity, it is definitely possible to create a software intense project. FPGA does however share the Arduino boards' problem when it comes to graphical output. It is possible to connect the board to a screen or a projector with a VGA-cable, but it would require a lot of tinkering to make it work and look good.

4.2.3 Raspberry Pi

The Raspberry Pi is a small computer. You can use basically any programming language you want with it. It also has an HDMI output so connecting it to a screen or a projector, and having it look good, is much easier than the other setups analyzed [11].

4.2.4 Conclusion

Of the different sensor setups analyzed the Raspberry Pi was selected. It is cheap and easy to use, but also powerful. The model selected for this project was the model 2B [12]. It has USB ports so you can plug in a mouse and keyboard, something the Arduino and FPGA boards does not have. It has general-purpose input/output (GPIO) pins for connecting sensors and buttons to it and, with the micro-USB port, an easy way to supply power to it from a power outlet. It also has an Ethernet port which makes it easy to connect it to the internet if required. And, as previously mentioned, and HDMI
output which makes displaying information on a screen or with a projector trivial.

The Raspberry Pi has all the advantages of the other two solutions, but none of the shortcomings. This made it an easy choice.

4.3 Programming Language

Since the programming was done with a Raspberry Pi, almost any language could be used. Python, C, C++, Java, Scratch and Ruby all come installed by default [13]. There are not many performance constraints thanks to the Raspberry Pi 2B’s 1GB primary memory and 900MHz quad-code CPU, so no language can be disqualified because of performance concerns. The program is going to have to display information on a screen so having some, preferably easy, way to make it look good is a necessity.

Java was selected as the programming language with Swing [14] toolkit used for GUI. Some library to have access to and to be able to control GPIO pins input/output is necessary and for this Pi4J [15] was selected. The Pi4J API gives the user full access to the I/O capabilities so that the sensor and buttons can be accessed and controlled by the Java code.

4.4 Selected texts

The program is supposed to show fitting sentences which matches the movement every time someone walks by and these sentences come from scripts of famous movies taken from www.imsdb.com. The movies currently used are the following: Star Wars IV, Star Wars V, Star Wars VI, Toy Story, The Incredibles, and James Bond Tomorrow Never Dies. Famous movies were selected in the hope that the users will recognize the sentences and scenes they are from.

Not all sentences from the movies are used. Only those which describe some sort of movement are used. To find these sentences the program uses a list of words. If a sentence contains at least one of these words, it can be used.

4.5 Word lists

There are two word lists. One is for the words used to describe the current movement and the other is for the words used to describe the change of movement.
4.5.1 Movement word list
Each row is divided into four parts, the first is the word and the other three explain the movement. Each part is divided by a “:”. Each movement part contains two values, the first is the lower limit and the second is the upper limit where “-1” means that there is no limit. If the movement is within all of the limits, the word is valid.

The first movement part denotes how many walks by per minute. The second denotes how many groups there are per minute where a group is defined by people walking close to each other (pass the sensor a few seconds, or less, apart). The third denotes the average time it takes for the people to pass the sensor’s area in milliseconds.

For example slow:0,0.125:-1,-1:3000,-1 means that for the word “slow” to be selected the following has to be true: People has to walk by, on average, not more than 0.125 times every minute (once every 8:th minute), any grouping and they have to be in the area, on average, for at least 3 seconds. I.e. people have to walk by rarely and slowly.

See Appendix A for the full word list

4.5.2 Change of movement word list
Each row is divided into two parts, the first is the word and the other explains the change of movement. As the movement word list, the row is divided by a “:” character.

The change in movement part is simpler than the regular movement part in the movement word list. It consists of only two parts. The first is change of number of people passing and the other change of their walking by duration. Both parts can only be one of three possible values: increase, decrease or irrelevant. The increase is defined by a “1”, decrease by a “-1” and irrelevant by a “0”.

For example, busier:1,0 means that for the word “busier” to be selected the following has to be true: More people has to walk by, on average, every minute. The pace they are walking at does not matter in this case. If the “0” would be a “1” instead, it would mean that the people would also have to walk, on average, at a faster pace.

See Appendix B for the full word list
4.6 Graphical feedback

The program has a graphical component which can be displayed with a monitor or a projector. It shows the selected sentences, both current and previous ones. The important word in each sentence is bold so that the users will get a better understanding why the sentence was selected. The previously selected sentences are also shown so the users can see the recent history of the movement in the area. All sentences also have a timestamp next to them so the users can see when the movement was active.

There are three different screens available to be shown by the program and any user can cycle between them by pressing a button. One of the screens shows the movement over the past 10 minutes, the second shows the movement of the past hour and the third shows how the movement is changing based on the 10-minute movement check. Information of which screen is visible is shown in the top left corner of the screen with a short text describing what is shown.

![Image](image.jpg)

*Fig 6. The toggle buttons are on the top-left of the monitor. The latest result is shown in the upper-middle part and in the bottom part is a list of earlier results.*
4.7 Data collection
The device was setup at the entrance to the Mobile Life Centre office for two weeks to collect data of the movement in the area. With that amount of data, the program could predict certain days by using the previous days as input. The device was mostly left alone to collect the data with someone checking it periodically to make sure it was still working correctly. The checks were done by seeing if was registering people walking by correctly.

Fig 7. Picture of the sensor setup taken in front of the sensor.

Fig 8. Sensor setup from above
5 Prediction

5.1 Data
To be able to accurately predict future movements the prediction algorithm has to know past movements. The movement is detected by a sensor and is stored in the device for future use.

The data collected is a list of elements each containing timestamps in milliseconds when the events happened and the duration of the event. Every time someone walks by the sensor it will add a new element to this list. The elements in the list are divided into days where the day contains all the events that happened that day and the times when the program was up and running.

If the program would be down for a few minutes due to a crash or if the program is being upgraded, that timeframe has no impact on the prediction algorithm. When predicting a future date at that timeframe that day is not used since any events that may have happened at that time are unknown.

5.2 Weights
As discussed in chapter 2, the more recent dates should impact the prediction result more than the older dates and dates which fall on the same weekday as the date being predicted should impact the prediction result more than the other dates. The related works use a decay factor and an algorithm called “Order-k Markov Algorithm” to achieve this. These solutions are described in more detail in chapter 2. In this project these two ideas are combined by only looking at the $k$ most recent days when predicting and having the more recent days impact the prediction more than the older days. This is done by assigning weight to the dates. The older the date is the less weight it will get.

A weight of 2 means that that day is used twice by the algorithm, a weight of 3 means that it is used three times and so on. The days which are close to the prediction date have high weights and the further back it looks the lower the weights get until they reach a weight of zero, which means that the day has no impact on the prediction.

One of the related works use different states for different kinds of movement, how this works is described in chapter 2. While this project doesn’t track specific movement, the idea of different states is still used by having the different days of the week count as different states. All seven weekdays are seen as different states where movement from the same state has a greater impact on the prediction than movement from the other states. The states, i.e. the weekdays,
are not separated completely. If a date falls on the same weekday as the date being predicted, i.e. if it is in the same state, it will have its weight increased. If they would be separated completely, it would leave very little data for the prediction algorithm to work with and it would also mean that the algorithm could miss certain patterns from the other days which might be important.

Since this project is setup in an office, there is a clear distinction between workdays and weekends. Because of this, the workdays and weekends states are completely separated from each other. The weekends have a weight of zero when the program is predicting a workday and vice versa.

5.3 Algorithm

When the algorithm is predicting a future date it looks at all recorded data, although some data can be so old that it has a weight of zero. It divides the days into intervals of the same size. The interval size can be set to any size, for example 5 minutes, 15 minutes or one hour. The program then calculates the average number of people who pass the sensor and the average pass duration for each of these intervals, and the standard deviations, and uses this as the prediction. Each interval prediction is calculated independently of the other intervals using the matching intervals, i.e. the same time frame, in the data.

The interval predictions are calculated by using the average of all dates in the data. If a date has a weight of 2 then the data from that date is used twice, if it has a weight of 3 then the data from that date is used three times and so on. These values are added together and the average is used as the prediction and the standard deviation from these values is the uncertainty of the prediction. Because of the weights there are many more values used for the prediction than there are dates in the data.
6 Results

The project resulted in a fully working application which measures the rate of which people are walking by and how fast they are going, i.e. the movement. Using this information, the application displays a sentence, which describes the movement, on a wall using a projector or on a monitor. Every time someone walks by the sensor or the movement changes the sentence is changed to a new, more fitting, sentence.

6.1 Prediction results

The program can predict the movement on a day by using the recorded data. The prediction uses the actual data, i.e. timestamps, not the describing sentences. The predicted date is divided into intervals, all of equal size, of any size the users decides. For example, if the user wants to see more broadly how the movement changes over the course of a day an interval size of one hour can be selected. If the user wants smaller intervals an interval size of 10 minutes can be used.

The program was setup for about two weeks to gather data, from Monday the 12th of October to Friday the 23rd of October 2015. The data gathered during these two weeks were used to evaluate the program. In this sections are a number of graphs showing results from the prediction algorithm compared to the actual data gathered by the sensor. The thin black lines over the prediction staples in the graph show how uncertain the algorithm is of the result. Since the prediction comes from using the averages of the days with certain days having a bigger impact, i.e. bigger weights, these uncertainties are the standard deviations of the averages.

Graph 1 and Graph 2 both show a prediction of how many people will be walking by the sensor and compares the prediction to what actually happened. Graph 1 shows the result of the prediction algorithm compared to the actual data using an interval size of 15 minutes. The date predicted was Friday the 23rd of October 2015 by using data gathered during the second week, i.e. from Monday the 19th to Thursday the 22nd of October 2015. Graph 2 shows the same prediction but using the data gathered from both weeks, i.e. from Monday the 12th to Thursday the 22nd of October 2015.
Graph 1. Measured data in blue. Predicted data in red with standard deviation. Interval size of 15 minutes. Predicted number of people walking by on Friday the 23\textsuperscript{rd} of October 2015 by using data gathered between Monday the 19\textsuperscript{th} to Thursday the 22\textsuperscript{nd} of October 2015.

Graph 2. Measured data in blue. Predicted data in red with standard deviation. Interval size of 15 minutes. Predicted number of people walking by on Friday the 23\textsuperscript{rd} of October 2015 by using data gathered between Monday the 12\textsuperscript{th} to Thursday the 22\textsuperscript{nd} of October 2015.
As shown in *Graph 1* and *Graph 2*, the standard deviations are usually very large. The standard deviations are larger in *Graph 2* than they are in *Graph 1*. Since *Graph 1* is based on only one week while *Graph 2* is based on both weeks, this means that the data gathered differentiates between the two weeks. The predictions are slightly higher when comparing *Graph 2* to *Graph 1*, meaning that there was more movement during the first week than the second. This might be explained by people being curious about the setup, and walking past it a few times to see its response. This curiosity would of course wind down as time went on since after a while everyone working there would get used to it. If this is the case, this would make the prediction algorithm overestimate the movement in the office, especially when using the first week to predict. However, as seen in the graphs, the predictions are usually underestimates.

There are big spikes around lunch time from 11:30 to 12:30 in the measured data, especially between 11:30 and 11:45. There seems to have been an unusual amount of movement around that time compared to the previous days. The movement dies down very quickly while in the prediction the movement is pretty steady during lunch time. Since the prediction is based on the previous days, this means that in the previous days people generally took lunch later and more spread out than on Friday the 23rd.

To see if the result from the prediction algorithm would be different, another date was chosen to be examined. The graph below, *Graph 3*, shows the prediction algorithm’s result compared to the measured data on Monday the 19th of October 2015 by using data gathered from Monday the 12th to Sunday the 18th of October 2015.
Graph 3. Measured data in blue. Predicted data in red with standard deviation. Interval size of 15 minutes. Predicted number of people walking by on Monday the 19th of October 2015 by using data gathered between Monday the 12th to Sunday the 18th of October 2015.

As seen in Graph 3 the prediction is more accurate compared to Graph 1 and Graph 2. There are however a few places where it wildly overestimates how many people will walk by. These predictions do also come with very big standard deviations, showing that the algorithm is very uncertain about the prediction. Since the data set is small for this prediction these wild overestimations can be explained by one day in the data the prediction uses having a lot of movement during those time frames. Why there are so much movement during such a short amount of time can be explained, as previously done, by curios people seeing the setup for the first time and deliberately walking back and forth to see the program’s response. It could also be a pure coincidence and that these spikes happen sometimes, which would be almost impossible to predict accurately. However, these wild overestimations will happen less frequently the more data the algorithm has to work with. By comparing Graph 3 to Graph 2, which uses both weeks as data, it is clear that the overestimations are smaller. In other words, that these big spikes in the data have less of an impact on the prediction results. If even more data would be used these spikes would have even lesser impact on the prediction’s result which would make the overestimations be even smaller. Unfortunately, due to time constraints, that amount of data could not be gathered during this project.
Graph 4 and Graph 5 show the prediction of the duration people are in the sensor’s area of the same day as Graph 1 and Graph 2 using the same amount of data. By knowing the duration they are in the area and since the sensor’s area is constant, the speed of which the people are moving can be calculated. Graph 4 is showing the prediction which uses data from Monday the 19th to Thursday the 22nd of October 2015, the same data duration as Graph 1 is using. Graph 5 is showing the prediction which uses data from Monday the 12th to Thursday the 22nd of October 2015, the same data duration as Graph 2.

Graph 4. Measured data in blue. Predicted data in red with standard deviation. Interval size of 15 minutes. Predicted the duration people were in the sensor’s area on Friday the 23rd of October 2015 by using data gathered between Monday the 19th to Thursday the 22nd of October 2015.
Graph 5. Measured data in blue. Predicted data in red with standard deviation. Interval size of 15 minutes. Predicted the duration people were in the sensor’s area on Friday the 23rd of October 2015 by using data gathered between Monday the 12th to Thursday the 22nd of October 2015.

The accuracy of the duration predictions is much higher than the number of people walking by predictions. Since the duration prediction does not care how many people are walking by, this can be explained by people normally walking at similar speeds, very rarely running. The people also seem to walk at a similar speed throughout the day, which makes the prediction algorithm job easier and also the result less interesting.

Graph 6 and Graph 7 shown below both show the same prediction with the same data as Graph 1 and Graph 2 respectively. The difference is that Graph 6 and Graph 7 have an interval size of 60 minutes instead of 15 minutes. Graph 6 is using data gathered between Monday the 19th and Thursday the 22nd of October 2015. Graph 7 is using data gathered between Monday the 12th and Thursday the 22nd of October 2015. Both graphs are showing the prediction of Friday the 23rd.
Graph 6. Measured data in blue. Predicted data in red with standard deviation. Interval size of 60 minutes. Predicted number of people walking by on Friday the 23rd of October 2015 by using data gathered between Monday the 19th to Thursday the 22nd of October 2015.

Graph 7. Measured data in blue. Predicted data in red with standard deviation. Interval size of 60 minutes. Predicted number of people walking by on Friday the 23rd of October 2015 by using data gathered between Monday the 12th to Thursday the 22nd of October 2015.

Graph 6 and Graph 7 are, as expected, very similar to Graph 1 and Graph 2. The standard deviations are larger in Graph 7 than in Graph 6 just as the standard deviations are larger in Graph 2 than in Graph 1. The same conclusion can be drawn why this is the case, namely that Graph 7 is based on both weeks and thusly uses more data than Graph 6 which is only based on the second week and that
the data between the two weeks are noticeably different. The spike between 11:00 and 12:00 is very noticeable is Graph 6 and Graph 7, just like the spikes in Graph 1 and Graph 2 around the same time. Again this can be explained by an unusually amount of people walking past the area compared to the previous days. The algorithm result shown in Graph 7 expects more movement than the result shown in Graph 6, just like the algorithm result shown in Graph 2 expects more movement than the result in Graph 1.

### 6.2 Summary

The algorithm can get a good understanding of how a normal work day looks, but cannot predict sudden spikes. Theses spikes can also make the algorithm overestimate future movement and thusly have a negative impact on the prediction. If these spikes are random or if there is no clear pattern to them the spikes would have less impact on the prediction algorithm the more data it uses. If the device would be setup under a long period of time the spikes would cause negligible overestimations. Unfortunately, due to time constraints, that amount of data could not be gathered during this project.

As seen in the graphs, the uncertainty, i.e. the standard deviations, in the predictions are fairly large. The graphs which use data from both weeks have higher uncertainty in them than the graphs which only use data from one week, this means that the data gathered differentiates between the two weeks. The predictions which use data from both weeks are slightly higher than the predictions which only use data from the second week, this means that there were more movement the first week than the second week. This might be explained by people being curious about the setup and walking past it a few times to see its response. This curiosity would of course wind down as time went on since after a while everyone working there would get used to it. If this is the case, this would make the prediction algorithm overestimate the movement in the office, especially when using the first week to predict. However, as seen in the graphs, the predictions are usually underestimates.

The speed predictions are rather uninteresting since it seems like people walk at roughly the same pace all the time, regardless what time it is and what day it is.
7 Conclusions

This thesis presents a way to monitor and predict movement. This thesis focused on an area in an office, but the same method can be used to set it up almost anywhere. The setup was up and collected data for two weeks, from Monday the 12th to Friday the 23rd. The prediction algorithm was evaluated by using parts of the data as input and predicting a date which had been measured. This way the prediction can be compared to the real life results. From the evaluation it can be seen that more data, i.e. a longer experiment time would have been helpful. Unfortunately, due to time constraints of the project, more experiment time was not possible.

The result evaluation shows that the algorithm can get a good understanding of how a normal work day looks, but cannot predict sudden spikes. These spikes can also have a negative impact on the prediction algorithm by making it overestimate future movement during the same time frame.

7.1 Discussion

A part of the project was to also give feedback to the people walking by of the recent movement trends in the area. This feedback was given by showing text describing the recent movement trends on a monitor. This had the effect of people being curious, especially at the start of the monitoring. The monitor was placed in a way so that people looking at it would not disturb the motion sensor, but the curiosity might have caused people to walk by the monitor a few extra times to see the feedback. Since the monitoring was in a small area in an office, the people it was monitoring would have been mostly the same during the entire two weeks of recording. This would naturally make the effects of the curiosity be smaller each passing day. This might explain why the recorded movement is higher in the first week than in the second week. This has of course a negative impact on the prediction algorithm by causing the prediction to be too high. If this reasoning is true then the two parts of this project, the prediction and the text feedback, are at odds with each other and it might have been best if the monitoring was done without the feedback. However, this curiosity would be negligible if the device would be setup under a long period of time.
8 Future Work

Since the algorithm is bad at predicting sudden spikes it might be a good idea to explore how to fix it. One possible way would be to try to find if there are any patterns leading up to a spike. If any can be found it is possible that a spike can be accurately predicted. Instead of predicting a whole day beforehand the algorithm would instead predict the day as it is happening. It could use the events that has already transpired during the day to predict the rest of the day. If, for example, it finds the movement recorded in the morning matches a certain pattern which will cause some unusual behavior, such as a spike, later in the day. The algorithm would be able to use this information to make a better prediction than it previously had. In other words: the algorithm would be able to correct itself in real time.

To be able to collect data of movement and predict future movement can be very useful in certain situations. One such situation is in the public transport system. Being able to predict when people use it and how many, can be used to find inefficiencies in the train and bus schedule. It can also be used to send extra trains or busses to places where the algorithm thinks there will be a heavy load on the transport system to avoid any annoyances, such as full trains, before they happen. Overall leading to a better and more pleasant public transport system.
References


Appendix A

Here is the list of the words used to describe movement. The explanation of how it is used is in chapter 4.5.1.

fast:0.4,-1:1,-1:1,2500
quick:0.4,-1:0,0.3:-1,2500
slow:0,0.125:-1,-1:3000,-1
slowly:0,0.125:-1,-1:3700,-1
quickly:0,4,0.3:-1,2500
often:0.5,-1:5,-1:1,-1
rare:0,0.1:-1,1,-1
rapid:-1,-1:1,1:1,2000
swift:-1,-1:0,1:1,2000
swiftly:-1,-1:0,0.1:1,2000
racing:-1,-1:1,1:1,2000
flashing:0.2,0.4:0.5,-1:-1,-1
posthaste:0.8,-1:0.3,-1,-1,2000
hasty:-1,1:0.3,-1,-1,2250
active:0.5,-1:0.5,-1:1,-1
inactive:0,0:-1,1:-1,1
passive:0,0.1:-1,1:-1,-1
sluggish:0,1,0.25:-1,1:-1,1
hurried:0,4,0.6:-1,1:-1,2300
unhurried:0.05,0.25:0,2:-1,3500,-1
tardy:0.05,0.2:-1,1:3000,-1
abrupt:0.1,1:0,0.05,0.15:-1,-1
energetic:0.4,-1:0.5,-1:1,2500
sudden:0.125,-1:0.05,0.15:-1,-1
suddenly:0.125,-1:0.05,0.15:-1,-1
busy:0.4,-1:0.5,-1:-1,-1
swamped:0.75,-1:0.3,-1:-1,-1
idle:0,0:-1,1:-1,1
quiet:0,0,1:0,0.3:-1,-1
silent:0,0:-1,1:-1,-1
run:0.4,-1:1,-1:1,2500
runs:0.4,-1:1,-1:1,2500
races:0.75,-1:0.3,-1:-1,-1
race:0.75,-1:0.3,-1:-1,-1
inching:0.05,0.2:-1,1:3000,-1
sweating:-1,-1:1,-1:1,2000
straining:0.4,-1:1,1:-1,1
goes:-1,1:-1,1:2500,-1
falling:0,0.2:-1,1:-1,-1
pauses:0.2,0.4:0.5,-1:-1,-1
jutting:0.3,-1:-1,1:2000,3000
stands:0,0.1:0,0.3:-1,-1
walks:0.3,-1:-1,-1:2000,3000
walk:0.3,-1:-1,-1:2000,3000
skis:0.75,-1:0.3,-1:-1,1
slaloms:0.3,-1:-1,-1:2000,3000
creeps:-1,-1:-1,1:3500,-1
freezing:-1,-1:-1,1:3500,-1
impatiently:0.4,-1:0.5,-1:-1,2500
hesitant:0.05,0.2:-1,-1:3000,-1
moving:0.3,-1:-1,-1:2000,3000
steps:0.3,-1:-1,-1:2000,3000
passing:0.3,-1:-1,-1:2000,3000
sweeps:0.3,-1:-1,-1:2000,3000
waits:0.1,0.3:-1,1
panic:1,-1:-1,-1:1,2500
running:0.4,-1:-1,-1:1,2500
sprinting:0.4,-1:-1,-1:1,2000
flying:0.4,-1:-1,-1:1,2500
rushing:0.4,-1:-1,-1:1,2500
crushing:0.3,-1:-1,-1:2000,3000
fighting:0.3,-1:-1,-1:2000,3000
blast:0.125,-1:0.05,0.15:-1,1
hops:-1,-1:-1,1:3000,-1
walking:0.3,-1:-1,-1:2000,3000
darts:0.4,-1:-1,-1:1,2500
urgent:0.4,-1:-1,-1:1,2500
leave:0.3,-1:-1,-1:2000,3000
wait:0.1,0.25:-1,-1:-1,1
swipe:0.3,-1:-1,-1:2000,3000
sliding:0.3,-1:-1,-1:2000,3000
approaching:0.4,-1:-1,-1:1,2500
stumbling:-1,-1:-1,-1:3000,-1
threading:-1,-1:-1,-1:3000,-1
rushes:-1,-1:-1,-1:1,2500
escape:1,-1:-1,-1:1,2500
slides:0.4,-1:-1,-1:1,3000
crashing:0.125,-1:0.05,0.15:-1,1
motions:0.3,-1:-1,-1:2000,3000
go:0.3,-1:-1,-1:2000,3000
beat:0.2,0.4:0.5,-1:-1,1
barging:-1,-1:-1,-1:1,2500
heading:0.3,-1:-1,-1:2000,3000
hovers:0.05,0.2:-1,-1:3000,-1
continuous:0.3,-1:-1,-1:2000,3000
proceed:0.3,-1:-1,-1:2000,3000
!
!:0.4,-1:0.5,-1:-1,1
Appendix B

Here is the word list to describe the change in movement. The explanation of how it is used is in chapter 4.5.2.

faster:0,1
quicker:0,1
swifter:1,1
hastier:0,1
busier:1,0
rarer:-1,0
falling:-1,0
tardier:0,-1
slower:0,-1
quieter:-1,0
same:0,0