An exploratory research of ARCore's feature detection

ANNA EKLIND

LOVE STARK
Abstract

Augmented reality has been on the rise for some time now and begun making its way onto the mobile market for both IOS and Android. In 2017 Apple released ARKit for IOS which is a software development kit for developing augmented reality applications. To counter this, Google released their own variant called ARCore on the 1st of march 2018. ARCore is also a software development kit for developing augmented reality applications but made for the Android, Unity and Unreal platforms instead. Since ARCore is released recently it is still unknown what particular limitations may exist for it. The purpose of this paper is give an indication to companies and developers about ARCore’s potential limitations. The goal with this paper and work is to map how well ARCore works during different circumstances, and in particular, how its feature detection works and behaves.

A quantitative research was done with the usage of the case study method. Various tests were performed with a modified test-application supplied by Google. The tests included testing how ARCore’s feature detection, the process that analyzes the environment presented to the application. This which enables the user of an application to place a virtual object on the physical environment. The tests were done to see how ARCore works during different light levels, different types of surfaces, different angles and the difference between having the device stationary or moving. From the testing that were done some conclusions could be drawn about the light levels, surfaces and differences between a moving and stationary device. More research and testing following these principles need to be done to draw even more conclusions of the system and its limitations. How these should be done is presented and discussed.

Keywords— ARCore; augmented reality; Android; feature detection; markerless tracking; Google
Abstract


Keywords — ARCore; förstärkt verklighet; Android; struktursdetektor; markörlös spårning; Google
Acknowledgements

We would like to thank Slagkryssaren whom were the stakeholders of the project and which enabled us to implement it and carry out the work with the thesis. We would especially like to thank our supervisor at Slagkryssaren, Oskar Henriksson, whom provided a lot of guidance and support in the development work. We would also like to thank our supervisor at KTH, Firdose Saeik, whom gave us guidance throughout the thesis. Finally, we would like to thank our examiner at KTH, Konrad Tollmar, for making this work possible.
## Contents

1 Introduction .............................................................................. 7  
  1.1 Background ......................................................................... 7  
  1.2 Problem .............................................................................. 7  
  1.3 Purpose .............................................................................. 8  
  1.4 Goal .................................................................................. 8  
    1.4.1 Social benefits, Ethics and Sustainability ......................... 8  
  1.5 Methodology ....................................................................... 8  
  1.6 Stakeholder ....................................................................... 9  
  1.7 Delimitations ..................................................................... 9  
  1.8 Disposition ....................................................................... 9  

2 Background theory ................................................................... 11  
  2.1 Augmented Reality .............................................................. 11  
    2.1.1 Trackers and Feature Descriptors .................................. 11  
  2.2 ARCore ............................................................................ 11  
    2.2.1 Motion tracking .......................................................... 12  
    2.2.2 Environmental understanding ..................................... 12  
    2.2.3 Light estimation ........................................................ 12  
    2.2.4 User interaction ......................................................... 12  
    2.2.5 Oriented Points .......................................................... 12  
    2.2.6 Anchors and Trackables .............................................. 12  
  2.3 Related work ..................................................................... 13  

3 Development/Models/Methods .................................................. 14  
  3.1 Empirical research .............................................................. 14  
    3.1.1 Case study ................................................................. 14  
  3.2 Software development ....................................................... 14  
    3.2.1 Scrum ..................................................................... 15  
  3.3 ANOVA ........................................................................... 15  
    3.3.1 T-test ..................................................................... 16  

4 Evaluation criteria .................................................................... 17  
  4.1 Light intensity ................................................................... 17  
  4.2 Surface ............................................................................ 18  
  4.3 Angle ............................................................................... 18  
  4.4 Motion ............................................................................ 18  
  4.5 Performance ..................................................................... 18  
  4.6 Battery ............................................................................ 18  
  4.7 Non-horizontal surface ..................................................... 19  
  4.8 Data collection .................................................................. 19  
    4.8.1 Detected points ........................................................ 19  
    4.8.2 Size of plane ............................................................ 19  
    4.8.3 Time ...................................................................... 19  
    4.8.4 Analysis of the collected data ..................................... 19  

5 Implementation ....................................................................... 20  
  5.1 Literature study ................................................................. 20  
  5.2 Case study ....................................................................... 20  
    5.2.1 Definition of objectives and planning ......................... 20  
    5.2.2 Data collection procedure ....................................... 20  
    5.2.3 Execution of data collection procedure ..................... 20  
    5.2.4 Analysis of collected data ....................................... 25  
    5.2.5 Presentation and conclusion of outcome .................. 25  
  5.3 Software development ...................................................... 25  
    5.3.1 Application .............................................................. 26  
    5.3.2 Added features ........................................................ 26  

6 Result .................................................................................... 27  
  6.1 Comparing light levels ...................................................... 27  
  6.2 Comparing angles ........................................................... 29  
  6.3 Comparing surfaces ......................................................... 32
1 Introduction

ARCore is a new software development kit released by Google with the intention of making it easier to create augmented reality applications. Applications developed through ARCore allows the user to interact with the environment, the technology overlays virtual contents onto the physical reality which we perceive with eyes, ears and other sensory organs. Augmented Reality has already been implemented within different sectors and categories such as industry, healthcare, education, marketing and entertainment, however the success of Augmented Reality depends on social acceptance and whether the technology is sufficiently user-friendly or not [1, 2].

1.1 Background

Augmented reality is the concept of combining a direct or indirect view of the real-world with virtual content, where the elements that are shown will be enhanced in some way or another. This enhancement is usually done via some graphical addition to the real-world projection shown. Unlike virtual reality which projects a completely virtual world to the user, augmented reality only alters the current perception of the world [3].

There are many different techniques in using augmented reality. The first functioning augmented reality system were invented by the U.S. Air Force’s Armstrong Laboratory in 1992 with their Virtual Fixtures system which were used to increase performance in operators [4]. The usage of augmented reality in modern phones has seen an increase the past couple of years, with Pokémon GO produced by Niantic and released in 2016 which turned out to be the top mobile game in the US [5].

The 1st of March 2018, Google released their new software development kit, ARCore, to be used when developing applications and games with augmented reality for Android. To enable virtual content appear in the real world seen through the user’s camera, three significant technologies are used by ARCore: (i) Motion tracking, (ii) Environmental understanding and (iii) Light estimation [6]. These are further described in section 2.2, ARCore. With these features ARCore could potentially place an object on your phone which integrates with the real world in a seamless way. An object could be placed virtually on a table and then the phone could be moved around in different ways and the object should still remain in the same position.

1.2 Problem

The follow-up question in this case is, how well does ARCore work. Since this is a new software development kit on the market, it is still pretty uncertain how good its functionality is. It is not known how great the user experience of ARCore is, nor is it known how well ARCore works for a developer using it, both in aspects of usability but also other limiting factors. It is not known about how performance demanding it is to use this kit either, will it be usable by older phones or will it demand more powerful solutions? How ARCore holds up for the user of an application is also not known currently. Are there any scenarios were it will work better than in other scenarios? Does the lighting conditions or different surfaces make a noticeable difference. Will objects still be present even if the phone is moved in a fast manner. There is also the matter of battery-time for a phone using an ARCore application. How much will it drain during usage and is it more than just using your camera.

All of these things are currently unknown to this present day and could be summarized to the different strengths and weaknesses that may or may not be present in the software development kit ARCore. The focus during this thesis will be to investigate ARCore’s feature detection. The process which enables the user of an application to place a virtual object on the physical environment and under what circumstances this is possible.

Within computer vision, the area of computer science which automatically processes and understands the content of an image, the execution of tasks such as image analysis and object tracking, rely on the occurrence of low-level features within an image. Wherein the low-level features can be specific structures which appear in the image, such as edges, objects or points. These points are commonly referred to as interesting points or feature points, hence the concept of feature detection. The feature points identified within an image may correspond to actual points present in the scene, they may also correspond to reflections or shadows that have arisen. The detection of feature points within a scene is affected by the light conditions in the surroundings, demanding light can therefore result in that no feature points are detected [7]. The way in which ARCore behaves under different light conditions will therefore be investigated, among other criteria.
To be able to identify whether the feature detection of ARCore works under certain conditions, an application will be developed through the software development kit. A sample project provided by Google will be used as foundation during the development of the application, whereby a big part of the thesis will be devoted to develop an evaluation model with relevant test criteria, and a way to interpret and analyze the received results and data. Thus, the main focus is not to create an application with a range of features, but rather to try out the technology of ARCore in order to map its limitations.

1.3 Purpose
The purpose of the report is to provide building blocks for other researchers to continue evaluating ARCore and its feature detection. With this thesis building blocks could be provided by showing how data could be collected and analyzed and how measurements could be done for different types of tests.

The purpose with the work is to give an indication to companies and developers of the current state of ARCore and the feature detection it uses. Whether companies or developers can make use of ARCore it could provide help by knowing any limiting factors when it comes to the feature detection. I.e. whether there are any situations where the feature detection works better or if there are anywhere it performs in an inferior way.

1.4 Goal
The goal with the work is to map how well-functioning ARCore's feature detection is as it is today. Whether the feature detection functionality performs different depending on varying environmental circumstances. Whereby the identified limitations hopefully can be useful for companies when developing augmented reality applications through ARCore.

1.4.1 Social benefits, Ethics and Sustainability
The possible success of ARCore and if the companies deem it as a useful tool in producing augmented reality applications for smartphones, social benefits could be seen to arise from this. Applications using ARCore could be used as instruction, i.e. by giving instructions to something you point your camera at in real-life. The application could then give you step-by-step instructions related to your surroundings, while giving you the ability to see it from different perspectives with your phone. Those instructions could include the building of a certain furniture from Ikeaf, assembling some sort of electronic product or even a medical intervention.

In regards to ethics with the usage of ARCore there is the issue with using the camera. This issue is present for both the developers and the users. For a developer the issue stems from the usage of what is shown on the camera, is that information saved in a non-ethical way or is it handled with discretion. This issue needs to be handled by the developers in an ethical way. For the users there is the possibility of a problem with the usage of the camera, if photos and videos are taken during moments and places where they should not. This issue is harder to tackle and goes back to the morals of each individual using the application.

A bigger spread of augmented reality for people could provide benefits to sustainability overall. With augmented reality the need for prototypes could decrease since you could superimpose a 3D-model of that prototype onto a device's screen so it could be inspected that way instead of having a need to produce a prototype that requires material of different sorts. Applications that give out immersive step-by-step instructions could reduce the need for technicians which in turn could reduce the resource consumption if a technician does not need to be sent out for repairs.

1.5 Methodology
In order to define the objectives of the thesis paper, specify the problem definition further and to provide a theoretical background of the problem area, a literature study was conducted at an early stage of the thesis. A literature study is a systematic way of reviewing literature in order to gain more knowledge of an area or subject, with a scientific purpose as basis. Conducting a literature study means systematic: (i) search/data collection, (ii) review and (iii) quality assessment of found data, to choose literature that seems relevant with the scientific purpose in mind. The relevance of which is determined by taking, among other things, publication year, problem definition, title and abstract into consideration [9].

A literature review was also made to find a suitable research method to work according to during the project, whereby the choice was made to carry out the research according to the strategy of a case study. A scientific method is a prerequisite in order to achieve the goals of a study or a project and to conduct
it well. One way of implementing research is through a case study, where a study of a specific case is conducted. By looking at an individual case and study things in detail, an in-depth understanding of the phenomenon intended to investigate, can be provided. The term case study includes the design of the research process to examine an issue and is suitable when the researcher conducting the study does not have full control over the phenomenon in its real-world context. The strategy of a case study allows usage of a variety of methods, which enables the researcher to choose appropriate method depending on the case studied and under what circumstances. A mixture of several data collection methods and different types of data is encouraged to promote the quality of the study and its outcome [10, p. 53-54]. The process of a case study is further described in subsection 3.3.1.

Enabling development work to proceed according to plan and to remain it under control, software development methodologies are applied within the process of developing software systems. Applying a methodology means to divide the process of developing a software system into different steps, phases and software development related activities [11, p. 103]. During the implementation phase of the application, an agile software development method was applied due to its dynamic nature. The company which we worked with during the implementation phase set us up to work with the project management application Trello, with an agile approach, similar to Scrum. Section 3.2 presents the agile way of working in more detail.

1.6 Stakeholder
The focus of this thesis was an initiative of the company Slagkryssaren. Slagkryssaren is a software development company who offer services within mobile and web development across all major platforms [12]. The research conducted and presented in this thesis intends to give an indication to Slagkryssaren of ARCore and its current limitations. The company had no further experience in developing augmented reality applications and were therefore interested in knowing more about where the technology is today and possible application areas. The reason why Google’s platform ARCore was chosen to develop through and investigated was due to our previous experiences and access to equipment.

1.7 Delimitations
The thesis has been limited to investigate ARCore’s functionality related to feature detection, the process of finding features within a picture. However, there are a range of other criteria (described in section 4.2) which could be significant when it comes to the usability of an Augmented Reality application and in interest for companies to know when deciding whether developing applications through ARCore is something to invest in or not.

A comparison between an application developed through ARCore and an application developed through ARKit, Apple’s software development kit for augmented reality applications, could have been valuable when evaluating how well functioning ARCore’s feature detection is. Also a comparison with other frameworks intended for Android development could have been of interest [13], due to time constraints this was not an option.

1.8 Disposition
The thesis is structured as follows. Chapter 2 gives the background theory related to the work. It describes augmented reality in its general terms and goes through the various trackers and feature descriptors that could be used in conjunction with augmented reality. This chapter also describes ARCore and the theory behind it in regards to its motion tracking, environmental understanding, light estimation, user interaction, oriented points, anchors and trackables. The last thing featured in this chapter is a section containing related work featuring scientific papers, which relate to this paper and were found during the literature study of augmented reality.

Chapter 3 describes the different work methods that were studied for this paper and how they work. Empirical research is described as is case study which is a part of empirical research. Different methods for software development is also described as is Scrum which is a popular method in software development.

Chapter 4 describes the different evaluation criteria that is present in the testing. The different evaluation criteria are as follows: light intensity, surface, angle, motion, performance, battery and non-horizontal surfaces, which also constitute all sections in this section. The data collection is described in what data will be collected and how it will be done.

Chapter 5 describes the implementation phase of the project. The literature studies that were conducted are described and detailed in a section. The case study that was implemented for this project gets its
description in its own steps. The software development method that was used during the project is described, how Scrum was used in this project and how the test application works.

Chapter 6 is the result chapter which presents all the results that were acquired during the testing phase. These results include comparing of the light levels, comparing of the angles and comparing of the sheets which are presented via graphs.

Chapter 7 presents the statistical analysis done on the resulting data. The first section presents the results of the ANOVA analysis of the acquired results. The second section does the post-hoc tests on those tests that showed to have a significant statistical difference.

Chapter 8 goes through what conclusions were reached during this project. The purpose and goals introduced in the introduction and are discussed whether there were any conclusions reached in those aspects. Other areas from the introduction are also discussed, included are the ethical aspect and the choice of methods. There is a section which analyzes and discusses the results presented in chapter 6 and what conclusions can be drawn from there. Sources of errors, delimitations and future work are also topics which are discussed in this section where sources of error discusses the possible ways that the results could have taken effect from outside errors whereas continued work discusses how this work and research could be continued for other researchers.
2 Background theory

This chapter briefly introduce some history of augmented reality, followed by a theoretical description of the technology used to locate the user of an application within the environment, the process of tracking. The second part of the chapter provides a theoretical presentation of the fundamental concepts behind ARCore, which illustrates how the technique of ARCore enables the experience of augmented reality through an application. Finally, some related work to the problem area is discussed.

2.1 Augmented Reality

As a concept augmented reality was talked about as early as 1901 by the author Frank Baum in his short story "The Master Key" wherein he discusses some sort of spectacles that overlays data onto real life, which he called a character marker. [14, p. 19] In 1968, Harvard professor Ivan Sutherland invents the head-mounted display which changes what is displayed based on where the user is gazing, based on head tracking [15]. In 1980, Canadian researcher and inventor Steve Mann invents the EyeTap. This device is a sort of wearable computer, designed to be used as a pair of spectacles. What these do is capturing the world via a sort of camera and then relays that video to the user in an enhanced way [16].

Augmented reality has made its way onto modern smartphones. Since the release of applications like Pokémon GO [5] and the inclusion of augmented reality functionality in applications like Snapchat there has been a surge in the interest in augmented reality for mobile applications and more companies have been trying out the technology [8].

2.1.1 Trackers and Feature Descriptors

It is important for augmented reality systems to be able to track their current positions in the real-world relative to the supposed objects on the screen with distance, rotation and direction taken into account. To do this AR applications and programs usually use trackers and feature detectors/descriptors. Marker tracking is one of those techniques used for tracking. It is used by placing some sort of object out in the real world. These objects could be a QR-code, a black cube drawn on a white paper or some other easily recognizable marker [17]. By having these markers the application has a reference in the real world in how it should superimpose 3D objects onto the scene in way of size and position. This gives an advantage in that when the image is analyzed it could search for the pre-defined easily recognizable markers which could lower the Central Processing Unit, CPU, usage and the overall demand of the system in comparison to not having a reference. A disadvantage of using marker-based tracking is that you are limited in what situations you can use the application since a marker always has to be provided one way or another.

Another technique that could be used for tracking is the markerless tracking technique. The difference to the marker tracking is that markerless tracking does not use a marker (hence the name) but instead via advanced algorithms try to evaluate the current environment presented to the camera. It does this by detecting certain features, like horizontal surfaces, corners and the likes [17]. An advantage to using markerless compared to marker-based tracking is that you open up the possibilities in where you can use the application since you are not limited to the placement of markers for the application to work. The disadvantage is that the feature detection can be very CPU intensive for the devices running the application and it is not certain that it can detect a surface during all kinds of conditions.

For detecting markers but also for detecting certain features, different feature descriptor algorithms could be used. Feature descriptors are a part of computer vision and the image processing fields. An image or video feed is analyzed, usually pixel-by-pixel, for certain features that could be found within the picture. The features that usually can be detected include edges, corners (interest points), blobs (regions of interest points) and ridges. The information collected during this analysis is then presented as distinctive, invariant image feature points, which easily can be matched between images to perform tasks such as object detection and recognition, or to compute geometrical transformations between images. This makes it possible to track these features over time which in turn makes it possible for an AR application to interact with these points, almost in the same way as with the tracker solution. Algorithms used as feature descriptors include Scale-Invariant Feature Transform (SIFT), Speeded Up Robust Features (SURF), oriented FAST, rotated BRIEF (ORB) and Fast Retina Keypoints (FREAK) [18, 19].

2.2 ARCore

ARCore tracks the position of the phone and in that builds its own understanding of the world around it. The motion tracking feature identifies interesting points, called features/feature points, and tracks how these points move over time. ARCore then determines the position and the orientation of the phone, which is also referred to as the pose of the phone. Since there is no need for a marker to be used during the usage
of ARCore it can be assumed that it is using a markerless tracking technique. It is currently unknown what and if ARCore uses a certain feature descriptor though, due to lack of information.

2.2.1 Motion tracking
ARCore detects with the camera so called feature points, which are visually distinct features found in the picture [6]. Through identification of feature points, together with the device’s orientation and its accelerometer sensors, ARCore is able to track the motion of the the device. The concept of motion tracking is implemented by using an algorithm known as visual-inertial odometry (VIO). VIO uses the internal motion sensors of the device to track the device’s position and orientation relative to where it started in combination with identification of image features from the device’s camera [20, p. 56].

2.2.2 Environmental understanding
ARCore learns of its surroundings by searching for certain feature points that form a cluster. It will then make these clusters available as a plane in the application. These are searched for on horizontal planes like tables, floors etc and enables the user to place virtual objects on flat surfaces [6].

While motion tracking uses VIO for identifying feature points to map and track the user’s position, the concept of environmental understanding makes use of it for identifying objects and their pose [20, p. 11]. ARCore provides the functionality of identifying planes and surfaces automatically, through meshing, which is the process of taking a cluster of feature points and construct a mesh from it. The mesh generated through this process is further shaded and rendered into the scene [20, p. 75-76]. Thus, to get an environmental understanding and to generate surfaces and planes, the techniques VIO and meshing are used by ARCore to provide desired functionality [20, p. 10].

2.2.3 Light estimation
ARCore has the ability to detect the light level of the current area. With the aim to make a more detailed representation of the 3D objects on the screen. I.e. make shadows and light appear on objects as if they were actually in the real-world [6]. To be able to replicate the light conditions within the scene, ARCore makes use of an image analysis algorithm. The current image of the device gets analyzed, an average light intensity is calculated and is further applied as global light to the 3D objects within the scene [20, p. 100].

2.2.4 User interaction
In order to let a user place an object on the scene, ARCore uses the technology of ray casting. When the screen of the phone gets tapped or some other kind of interaction between the screen and a user occurs, a point with two dimensions corresponding to the device’s screen is generated and a ray is projected from this point into the camera’s view of the world, the scene. If the ray intersects with some real-world geometry, represented by a plane or an oriented point, the position and orientation of the intersection is returned. Intersections are sorted by depth, thus only the closest intersection with a plane or an oriented point is considered [20, p. 77-79].

2.2.5 Oriented Points
ARCore provides functionality that lets the user of an application to place virtual objects on non-horizontal surfaces through oriented points. When a user wants to place an object on a non-horizontal surface, the same procedure as described in subsection 2.2.4 follows. The position and orientation of the intersection will be returned, and ARCore will make use of this feature point’s neighbors to try to estimate the angle of the surface at the intersection. ARCore will then utilize this angle to generate and return a new position and orientation, which constitutes an oriented point and enables the attempt to place an object on a non-horizontal surface [6].

2.2.6 Anchors and Trackables
To track a placed 3D object over time ARCore uses something called anchors. Anchors track the orientation and position in the real-world of the 3D objects and represents an attachment point. These anchors are attached to something called trackables. Trackables can both be points and planes and ARCore will track them over time. The creation of an anchor is usually based on the pose of an intersection generated when a user attempts to place out an object on the scene, as described in subsection 2.2.4, User Interaction [6][20, p. 79].
2.3 Related work

When evaluating an application and investigating its functionality, similar work done can be valuable in the process of finding out what to evaluate and how to interpret the results of the evaluation. During the literature study conducted at an early stage, some similar studies of interest were found.

The paper "Evaluation of Augmented Reality Frameworks for Android Development" [13] investigates different frameworks for Android development, as the title tells. With the aim to illustrate that the choice of appropriate framework when developing augmented reality applications depends on which context the application will be used within. By defining a set of constraints as a part of the evaluation of the frameworks, the researchers were able to show whether one framework outperformed another one. During their work with the evaluation, they defined some scenarios, which consisted of a collection of constraints, to reflect different use cases. The use cases enabled a breakdown of when to use which framework, for instance the framework metaio was the most appropriate one for the scenario of an Interior Design application. This way of presenting the findings of an evaluation seems quite powerful. The information could be valuable for companies which receive requests to develop augmented reality applications for certain purposes, there may not be time for them to perform an evaluation like this by themselves. It could also be useful for researchers or other professionals, in order to conduct further work within the area of developing augmented reality applications.

The publication "Checklist to Evaluate Augmented Reality Applications" [2] presents a way to evaluate Augmented Reality applications in terms of usability. A checklist was developed for an augmented reality context, with lack of such evaluation methods within the area as incentive. The most interesting part of the paper is the conclusions, whereby the attention is drawn to the need of an agreement of "a conceptual definition of quality as well as a set of quality criteria, that can be implemented using a checklist". To state whether an application works well or badly, seems like quite a struggle, with the definition of "well" in mind. It is also mentioned in the paper that the main limitation of the study was the fact that only two applications were evaluated, which indeed makes it hard to ensure that the evaluation model can be generalized and used when evaluating a variety of augmented reality applications. These types of insights are great to have in mind when evaluating the results of a study. However, in further research their plan is to improve the checklist by applying it to additional augmented reality applications, which seems like a reasonable way to ensure the quality of it.

Another related work found was the paper "Feature Point Detection under Extreme Lighting Conditions" [7]. An evaluation of four different feature point detectors has been made, under extreme lighting conditions. In order to do the evaluation, experiments were conducted, with changes of camera viewpoint, camera distance and scene lighting. Two different test scenes were used during the experiments, one planar 2D scene containing different posters and one 3D scene containing a number of solid non-planar objects. The main approach of the paper was to investigate the improvement of feature point detection when using the camera technique High Dynamic Range, in comparison with the traditional Low Dynamic Range technology. HDR is capable of representing a infinite light intensity range(in theory), while LDR is limited to an intensity range of 0-255, which constitutes the difference between the two techniques. The outset of the paper "Feature Point Detection under Extreme Lighting Conditions" differs from the problem definition discussed in this thesis, however the way the experiments were conducted is still relevant for this thesis and paper.
3 Development/Models/Methods

This chapter introduces empirical research, followed by a description of how to conduct a case study and a theoretical background of agile software development. Lastly a statistical way of analyzing data is presented.

3.1 Empirical research

Empirical research originates from the theory empiricism, where knowledge about an area is obtained through experience and observation [21]. To be able to evaluate and validate the results of a research, empirical methods such as controlled experiments, case studies, surveys and analyses, are used as tools. There is a need of these methods to be able to scientifically decide whether the outcome within a research, aimed to be analyzed, is evaluated as great or bad and to derive a meaningful results. Exploratory and descriptive research are two ways of implementing empirical research, where the data collected throughout the study can be either quantitative or qualitative [22, p. 7-8].

Exploratory research is used for problems that have newly arisen and have not been studied in any great length before. It generally is not meant to draw definitive conclusions but rather serve to help us understand the problem better, almost in the form of a pre-study [23]. This to generate a new hypothesis regarding the problem that can be further researched on at another time [24, p. 135]. Descriptive research, on the other hand, aims to describe a current problem or situation further with collected data, rather than exploring a new and unknown area [25, 26].

The data collected within quantitative research is numerical in form of numbers and statistics. Gathered data is unchanging, thus the study can be replicated several times and the outcome of it will remain. The main goal with quantitative research is to generalize gathered data across a population or to explain a certain phenomenon [26].

In contrast to quantitative research, the qualitative approach is based on data consisting of interview's which describe and capture people's personal perspective of the subject in question. The aim with qualitative research is to get a more realistic depiction of the world which can not be obtained in the same way by using numerical data and analyzing statistics. A certain phenomenon in this case, is read by the researcher based on people's opinions and interpretations, rather than explaining it with numerical data generalized across a population [27, 26][22, p. 8].

3.1.1 Case study

Case study is an example of an flexible empirical method and is well suited for studies within software engineering. The method is primarily used for exploratory investigations, and is applied by researchers and professionals within the area to understand, explain and prove the power of a new technique, method, tool or technology. It explains, in a scientific manner, how the study will be conducted. A case study of an exploratory approach has a research question as starting point, whereupon data is collected and analyzed in order to find an answer to the question [28, 24].

According to the guidelines in Case Study Research in Software Engineering [29], following are the major steps going through within the process of a case study:

- (i) definition of objectives and planning of case study
- (ii) data collection procedure is set
- (iii) execution of data collection procedure
- (iv) analysis of collected data
- (v) presentation and conclusion of outcome

In the end, there are three main things that characterize a well-conducted case study: (i) the case study is flexible when it comes to its design, to cope with phenomenon within software engineering, (ii) the conclusion of its outcome is based on evidence, like quantitative or qualitative data, which has been collected from a number of different sources in a systemically manner and (iii) the study adds something to already existing knowledge, by building new theory or by being based on previously established research and theory [29, p. 19].

3.2 Software development

There are a variety of different models and frameworks to choose among and work according to, the traditional waterfall method is one of them. The waterfall method is an engineering method used in software
development. It works by being a linear way of working with different stages of development. These stages include Requirements, Analysis, Design, Coding, Testing and Operations. You can only progress from one stage if it is considered to be fully complete [30, p. 31].

However, over the past years, the traditional waterfall model has declined to its extent, while agile software development has grown [30, p. 34]. Since the requirements of a product/system are updated continuously, a process model that allows unpredictable events is required when working with software development. A model where the software is not delivered as a big unit, but rather several subunits, versions, that are developed with further functionality over time [30, p. 57].

Agile methods work according to this model, stepwise and iterative software development where new releases are made available for the customer continuously. The customer is thereby involved in the process of developing the product/system, which means periodically feedback from the customer to the team of developers and updated requirements of the product. This prevents the system from losing its value when the final product is released [30, p. 58].

3.2.1 Scrum

Scrum is an agile methodology for development team, which provides room for independence, handling unpredictable events and solving complex problems. It is designed to maximize the flexibility, creativity and productivity of the development team [31, p. 3]. The methodology consists of a number of predefined activities, to create a working structure and to reduce the risk of unscheduled meetings. These activities have set time frames and serve different purposes. The basic idea with the activities is to: (i) evaluate work, (ii) plan future work and (iii) adapt work methodology according to evaluation of previous work.

The starting point of planning the project is to establish a product backlog, which is basically an ordered and prioritized list of the work that needs to be done in order to deliver the desired product/release. The product backlog is dynamic and changes over time as new functionality is added to the product, if further needs are identified in order to deliver a valuable product or if potential bugs have arisen that need to be addressed [31, p. 15][30, p. 73].

The heart of Scrum and its most innovative feature is the sprint. A sprint is a time period of maximum one month where the team work on developing a deliverable product/release of software, which is then, during the next sprint, developed further with new functionality. After one sprint has ended, the next one begins [31, p. 9][30, p. 73]. A brief description of the predefined activities that apply when working according to Scrum follows below.

(i) **Sprint Planning**, a sprint goal is set up and the functionality aimed to be developed during the current sprint is identified [32, p. 16].

(ii) **Daily Scrum**, a short meeting of fifteen minutes where the development team go the team around and discusses the following questions. What did I do yesterday to reach our sprint goal? What can I do today to help the team reaching our sprint goal? Are there are any issues that could prevent us from reaching our sprint goal? [32, p. 74]

(iii) **Sprint Review**, an occasion where the team shows what they have accomplished during the sprint to external parties, stakeholders. Two-way communication which results in ideas that could be included in the next sprint [32, p. 80].

(iv) **Sprint Retrospective**, the team analyzes the sprint that has passed and discusses the following questions. What went well? What did not go that well? What could be improved to the next sprint? [32, p. 84]

3.3 ANOVA

Analysis of variance, ANOVA, is a collection of statistical methods which are used to investigate whether the mean value of a variable differs between groups. This tool is used when mean values are being analyzed between more than two groups. The ANOVA tests the hypothesis which states that all mean values are the same, the null hypothesis. Whereby the alternative hypothesis states that there is a difference in the matter of the mean values between the groups which are being tested. The null hypothesis can be rejected if the ANOVA gives a significant result, and thus ensure that at least one of the groups mean value deviates [33, 34].

The spread of measurement values, the variation, is a key term within ANOVA. The variation is divided into two parts, wherein the first component indicates the variation between the groups, to what extent the mean values differ. The variation between groups is calculated by looking at each group’s mean value and compare it to the total mean, the average of all measurements. The variation within the groups is the other component that needs to be taken into account when doing an analysis of variance. It is calculated
by looking at each individual measurement within the group and compare it to the mean of the group, examine to what extent the value differs from the group’s mean [35, p. 361-364].

The comparison between the two components within ANOVA, the variation among the groups respectively the variation within the groups, results in a quota which is referred to as F-value or F-ratio.

\[ F = \text{the variation between groups} / \text{the variation within groups} \]

In order to decide whether the differences between the groups are of statistical significance, the F-value is compared with a critical value of F which states how great F must be in order to reject the null hypothesis. The critical limit depends on (i) the number of measurements which are done within the study (sample size) and the number of different groups which the measurements are done within, referred to as degrees of freedom and (ii) the probability of rejecting a null hypothesis when it is true, the significance level [36][35, p. 364]. The null hypothesis can be rejected if F shows to be greater than the critical value of F, and it can thereby be stated whether there exists a statistically significant difference between the means of the groups or not [34]. However, in order to distinguish which groups differ in their means, additional tests need to be conducted, so-called post-hoc tests. Conducting a post-hoc test basically means performing a test after the actual study has been conducted, wherein a t-test is an example of a post-hoc test [35, p. 523].

### 3.3.1 T-test

A t-test can be used for the purpose of investigating whether the means of two groups differ in a statistically significant manner, similar to the purpose of ANOVA, which is done by conducting a two-sample t-test [37]. Several t-tests can be conducted in order to map which groups have means that differ and thereby provide meaningful data to the one conducting the study. Similar to the F-value of ANOVA, a t-value (T) is used within the procedure of a t-test, wherein the difference between the means of paired groups is compared and the variation within a group is taken into account [35, p. 372].

From the t-value together with the degrees of freedom, a p-value can be retrieved, wherein the p-value is a measurement on how likely it is that the null hypothesis is true for a given test and data. A low p-value indicates that there is a low probability that the null hypothesis is true. The p-value is compared with the chosen significance level, if the p-value shows to be below the significance level the null hypothesis can be rejected. The null hypothesis which states that there is no statistically significant difference among the means of the two groups included in the study [38, 39].

However, when conducting several statistical tests and multiple comparisons, some tests will result in p-values less than the significance level alpha by chance, thus an increasing number of null hypothesis could be rejected even though they are true. One way to deal with this problem is to use the method of Bonferroni Correction, which takes this scenario into account. Instead of using the chosen alpha in order to determine whether the null hypothesis should be rejected or not, a lower critical value is applied, the Bonferroni correction of alpha. Wherein the Bonferroni correction of alpha means that alpha is divided by the number of data sets which are compared within the current test, thus if the p-value is less than the Bonferroni correction of alpha, the null hypothesis can be rejected [40, 41].
4 Evaluation criteria

This chapter presents how the data collection procedure was planned to be implemented. Section 4.1-4.3 presents the procedure of the tests that were performed during this research. Section 4.4-4.7 presents the procedure of the potential and possible tests which could be done in the future, but were postponed due to time constraints. In section 4.8 it is described how the data collection worked during these tests and also how the collected data was aimed to be analyzed.

4.1 Light intensity

The application was tested in how it works during different lighting conditions, following what I. Marneu et al. [13] did in their testing. The current light intensity was measured with an instrument. To determine the general lighting of the test area, the area was divided into equally sized squares. The light intensity was measured in the middle of each square, and an average light intensity of the area was calculated from these values. The recommendations of "Lighting Assessment in the Workplace" was followed, wherein areas less than 50 square meter are divided into a minimum of sixteen equally sized squares [42, p. 7-8].

<table>
<thead>
<tr>
<th>Type of day</th>
<th>Lux</th>
</tr>
</thead>
<tbody>
<tr>
<td>Very Bright Summer Day</td>
<td>100,000 Lux</td>
</tr>
<tr>
<td>Full Daylight</td>
<td>10,000 Lux</td>
</tr>
<tr>
<td>Overcast Summer Day</td>
<td>1,000 Lux</td>
</tr>
<tr>
<td>Very Dark Day</td>
<td>100 Lux</td>
</tr>
<tr>
<td>Twilight</td>
<td>10 Lux</td>
</tr>
<tr>
<td>Full Moon</td>
<td>&lt;1 Lux</td>
</tr>
</tbody>
</table>

Table 1: Examples of outdoor light levels. [43]

<table>
<thead>
<tr>
<th>Type of indoor environment</th>
<th>Lux</th>
</tr>
</thead>
<tbody>
<tr>
<td>Performance of visual tasks of low contrast and very small size for prolonged periods of time</td>
<td>2000-5000 Lux</td>
</tr>
<tr>
<td>Detailed Drawing Work, Very Detailed Mechanical Works</td>
<td>1500-2000 Lux</td>
</tr>
<tr>
<td>Normal Drawing Work, Detailed Mechanical Workshops, Operation Theaters</td>
<td>1,000 Lux</td>
</tr>
<tr>
<td>Supermarkets, Mechanical Workshops, Office Landscapes</td>
<td>750 Lux</td>
</tr>
<tr>
<td>Normal Office Work, PC Work, Study Library, Groceries, Show Rooms, Laboratories</td>
<td>500 Lux</td>
</tr>
<tr>
<td>Easy Office Work, Classes</td>
<td>250 Lux</td>
</tr>
<tr>
<td>Warehouses, Homes, Theaters, Archives</td>
<td>150 Lux</td>
</tr>
</tbody>
</table>

Table 2: Examples of recommended indoor light levels. [44]

Three different light levels were measured and then the application was tested. The first light level was of a dimly lit room with a lux around 50-120. The second light level was of a light room with a lux of around 400-800. The last light level was outside in the sun, at around 70,000-100,000 lux. The reason for the variations in the lux is because there was no available access to a room where the light can be controlled to the fullest extent. The application was tested against the same surface at the same angle and distance. It was to be seen if it is possible for ARCore to detect the same surface at different light levels. These tests were done to evaluate if the application was to be more suitable in certain lighting conditions. Whether an object could be placed or if there is a limit on how dark or light the general lighting could be. A lighting test was considered to have passed if it could produce a plane at the same position as
before. The size of the plane that is produced by the application was measured in its size by the functions Plane.getExtentX and Plane.getExtentZ. These measurements were done to determine if there were any size-differences within the different light levels.

The camera was pointed at a surface in the chosen angles, the phone was placed in a stand to remain in the same position. This was done for a set amount of time and was repeated until desired amount of test data was obtained. A motion test was then performed by holding the device perpendicular to the surface and dragging it forward and back. The application was then closed and the light intensity was changed. The procedure was performed for each desired light intensity.

4.2 Surface

A number of different surfaces were tested to examine how big of an impact the texture of a surface had on the application’s ability to understand the environment and detect feature points [6]. If the application could produce a plane at a surface it had passed the test with that certain surface.

The camera was pointed at a certain surface and was held in the chosen angles with a stand. This was done for a set amount of time and was repeated until desired amount of test data was obtained. A motion test was then performed by holding the device perpendicular to the surface and dragging it forward and back. The application was then closed and the surface was changed. The procedure was performed for each chosen surface. Four different surfaces in total were tested. For the indoor testing a certain carpet found in the office, a blank white sheet and a paisley duvet were used for the testing. For the outdoor testing the blank white sheet and the paisley duvet were used, to be able to compare the indoor results with the outdoor results. A test against grass was also done.

4.3 Angle

The application’s performance and ability to detect feature points could prove to be dependent on the angle the device is in [13, p. 38][7, p. 138]. This was tested, to see whether there are any angles that produce better results in the matter of detecting feature points and surfaces. The camera was pointed at a surface was held in a stand with one of the chosen angles. This was done for a set amount of time and was be repeated until desired amount of test data was obtained. No motion test was done with where angle was examined since we could not find a good way to perform those tests. The application was then closed and the angle was changed, the procedure was performed for each chosen angle.

4.4 Motion

A potential testing for motion could be to start with a slow movement and progress to a more rapid approach and examine whether: (i) the application could get an environmental understanding and (ii) possibly virtual content would get lost or not [45, 46]. An evaluation of the recovery time for the application could also be made to examine the case of rapid movement and coverage of the phone’s camera [47].

4.5 Performance

A potential testing of performance would be to place out different number of objects out on the scene to examine whether the performance decreases with increasing number of objects. It could be investigated if there exists a threshold for the number of objects placed out on the scene simultaneously that would cause the application to shut down. The Developer Guide for ARCore recommends to detach unused anchors to avoid decreased performance, usage of more than a dozen anchors could reduce the performance of the application significantly [48]. As it is today in the sample project provided by Google, hello_ar.java [49], each object gets its own anchor, and there is a limitation of twenty objects present at the scene at the same time set, to avoid overloading the rendering system and ARCore.

4.6 Battery

Augmented Reality applications deal with intensive computations and are sensitive in the matter of delays. A low end-to-end delay is required in order to sustain the application’s interactive functionality [50, 51]. Both these factors contribute to the battery lifetime of a phone in a negative manner. An examination of how quickly the AR-application drains a phone’s battery could therefore be made. Whether different number of objects placed out on the scene simultaneously have any impact on the battery could be investigated, with one, five, ten and twenty objects as starting point.

The battery usage of the phone in normal use and the battery usage of the phone when the camera is running for one minute respectively could be noted down. The application developed through ARCore
could then run for one minute and the battery usage could be noted down. The test could be performed with a certain amount of objects present on the scene at the same time and be repeated until desired results are obtained.

4.7 Non-horizontal surface

Taken into consideration that ARCore uses Oriented Points to enable the user of an application to place out objects on non-horizontal surfaces [6], following scenarios could be tested to examine the application: (i) a real object placed out on the scene and (ii) a surface perpendicular to an already detected surface (wall perpendicular to floor).

4.8 Data collection

During these tests various types of data was collected. This data was collected during a certain amount of time. This data was collected by the application and saved as .json files (JavaScript Object Notation) [52]. These files were interpreted by a java-program to determine the mean-value of a certain number of tests. The same test/criteria were replicated several times under the same circumstances to ensure that retrieved data is meaningful and complete [53]. A mean value was calculated from the collected data, across the set of tests that were performed, to enable comparison between various environmental conditions. During the tests notes were made of anything that could make any of the data deviate from the expected results to avoid faulty data.

4.8.1 Detected points

A measure on how many feature points are detected when the application is running was performed. This was done to see if there were any differences in detected points under different conditions. This was measured in number of feature points. The data was presented in number of detected points per second respectively total amount of points detected during the runtime of the test.

4.8.2 Size of plane

When a plane is created it will be of certain size. This size is specified by X and Z values. The function (Plane.getExtentX() and Plane.getExtentZ() respectively) returns the length of this plane’s bounding rectangle measured along the local X-axis of the coordinate space centered on the plane and the other for the corresponding Z-axis [54]. This was done to evaluate if there were any differences to the size when there are different conditions for ARCore. The data was presented in size per second and total size of the plane accomplished during the runtime of the test.

4.8.3 Time

The time for how long it will take for the application to detect a plane was measured. The time was measured in milliseconds and was measured by taking the time in milliseconds from when a plane is first detected minus the start time of when the benchmark started.

4.8.4 Analysis of the collected data

The data that was collected was looked at in different ways to determine how ARCore’s feature detection behaves in different situations and what the differences were. It was looked at how long it took for ARCore to detect a plane during certain conditions. These times were compared to each other to see if they differ greatly under differing circumstances and where a shorter time is considered better than a longer time. The sizes of these planes were also compared to each other from the different tests, where a larger plane was to be considered better than a smaller one. The amount of feature points detected was looked and a comparison was done. This test differs in the way that points in the PointCloud will join a plane when it is created, which means that there could be less points returned when there is a plane present in comparison to a test where no plane is present. This was looked at case-by-case and combined with the other data (i.e. if a plane was present when there was a decrease in points).
5 Implementation

This chapter aims to describe the case study conducted during the thesis, which implies the implementation of the literature study, the definition of criteria to be tested, as well as the development phase of the application. The work conducted during the thesis was distributed equally, both in terms of report writing, the implementation of the study and the software development.

5.1 Literature study

In order to gain knowledge about the area of augmented reality and ARCore, a pre-study was conducted. Literature was searched for and read through. The search was made mainly within the following databases of scientific publications: (i) IEEE Xplore, (ii) Google scholar and (iii) ScienceDirect. Scientific material was searched for using the keywords: augmented reality, limitations, evaluation, mobile, application. The material of relevance for the study was selected by taking title, abstract and publication year into consideration. Some of the publications found and reviewed during the pre-study, constitutes part of the thesis’s reference list. Additional material was found through the publications reference lists, as well as through recommendations of similar articles provided by the databases.

The fact that ARCore was recently released, made it a bit harder to find scientific material related to the technology which ARCore makes use of. At an early stage during the thesis, the only information which was available was found on Google’s website about ARCore. However, later on the book “Learn ARCore- Fundamentals of Google ARCore” [20] was published, which provided a more in-depth description of the technology behind ARCore and also guidance in developing applications through it.

Another literature study was conducted in order to select the appropriate research method to work according to. The keywords used were: software engineering, software development, research method. The search was conducted within the same databases as mentioned previously and with the same examination strategy in order to identify relevant material. The literature study continued as the thesis progressed, making more specific searches of current problem areas or subjects.

5.2 Case study

The study was basically conducted according to the five steps described in subsection 3.1.1, Case Study. Whereby the main focus initially was to define the objectives of the study and unravel the problem definition. Thereafter, the procedure of data collection was planned and set to enable the execution of the tests.

5.2.1 Definition of objectives and planning

The aim was initially to identify and describe the strengths and weaknesses of ARCore, however it was soon realized that having such a broad field of study would not benefit the outcome of the thesis. Thus, the objectives of the study came to focus on investigating ARCore’s feature detection. The case study was planned and the starting point of the study was together with the company which provided guidance in the development of the application and about the design of the tests. The process of developing the software is further described in subsection 5.3.

5.2.2 Data collection procedure

A literature study was conducted in order to define evaluation criteria, which would help to identify the limitations of ARCore and mainly its feature detection. The related work found during the literature study(subsection 2.3) was valuable when specifying the data collection process. By examine the API of ARCore, quantitative data(information about the scene seen through the phone) that could be extracted from the library was identified. The conditions for which data should be collected were decided, also details such as number of times each test should be executed and the frequency of reading and saving data. This step, planning the data collection, resulted in a protocol and a definition of the data collection procedure, which can be found in chapter four, Evaluation criteria. The three following evaluation criteria were chosen to proceed with during the project. They were chosen in consultation with the company and what they desired to get out of the evaluation: Light intensity, Surface and Angle.

5.2.3 Execution of data collection procedure

The tests run through the application were performed in a room with a window that was available at the company. The size of the room was measured in order to calculate the general lighting of it and the general lighting was measured and calculated right before and after every test run. The object which was highest in the room was a table with a height of 90 cm, therefore the measurement of the light intensity
was performed at this height for each and every measurement point. One test run involved selection of one surface which was then tested at different angles. The angles chosen to be tested were 90 degrees, 65 degrees and 45 degrees. These tests were carried out with the phone in a stand stationary, however an observation was made that the number of feature points detected was quite few and no planes were detected at all, which resulted in two criteria being added to each test run: (i) movement of the phone back and forward by holding it in the hand and angle the phone from approximately 45 to 85 degrees relative to the surface and (ii) movement of the phone by stretching out the arm back and forward with the phone perpendicular to the surface. Below follows a description of how the tests were conducted, also pictures which demonstrate how the scene looked like at different angles, see figure 1-3.

(i) Choose surface
(ii) Measure the general lightning
(iii) Choose for how long the measurements shall be conducted (10, 20 or 30 seconds) and how often the data shall be saved (two times per second, every second or every other second) in the settings menu of the application
(iv) Place the device in a stand
   a. Adjust the stand to an angle of 90 degrees
      i. Perform the measurements/the test -> repeat for five times
      ii. Apply the program which calculates the average between the five test runs on generated data
   b. Adjust the stand to an angle of 65 degrees
      i. Perform the measurements/the test -> repeat for five times
      ii. Apply the program which calculates the average between the five test runs on generated data
   c. Adjust the stand to an angle of 45 degrees
      i. Perform the measurements/the test -> repeat for five times
      ii. Apply the program which calculates the average between the five test runs on generated data
(v) Motion test
   a. Hold the device perpendicular to the surface and stretch it back and forward
      i. Perform the measurements/the test -> repeat for five times
      ii. Apply the program which calculates the average between the five test runs on generated data
(vi) Measure the general lightning
   \[\Rightarrow\] Repeat this procedure for desired surfaces and lightning conditions

Figure 1: 45 degree angle
The different surfaces chosen to investigate were a carpet which the room was covered in, a white sheet and a paisley-pattern duvet. The testing of the application was also carried out outdoors, due to the limited lighting conditions indoors. Grass was chosen to be the third surface to be tested outside, with the non-movable carpet in mind. These surfaces were chosen since they were determined to be common surfaces to be used by someone using an augmented reality application. There was not the same opportunity to perform the tests in an isolated environment outdoors, therefore only the tests that involved movement were executed. The following pictures show the surfaces which were used for the tests.
Figure 4: Carpet surface

Figure 5: Sheet surface

Figure 6: Paisley surface
In addition to the quantitative data collected during this procedure, some observations were also made. These are described in chapter seven, Conclusions. Below follows tables with the light intensities measured before and after each test run. The lighting measurements of the tests which were performed indoors are presented in table 3 and 4, followed by table 5 which presents the lighting measurements of the tests that were executed outdoors. Information regarding which light level corresponds to which light intensity is also listed, wherein light level 1 corresponds to a very dark day. Light level 2 reaches from the lighting present at indoor environments such as normal office work to the lighting found in a supermarket or within office landscapes. Light level 3 corresponds to a very bright summer day, which is found outdoors.

<table>
<thead>
<tr>
<th>Surface</th>
<th>Light intensity before the test run</th>
<th>Light intensity after the test run</th>
<th>Light level</th>
</tr>
</thead>
<tbody>
<tr>
<td>White sheet</td>
<td>88 Lux</td>
<td>127 Lux</td>
<td>Level 1</td>
</tr>
<tr>
<td>Paisley duvet</td>
<td>116 Lux</td>
<td>115 Lux</td>
<td>Level 1</td>
</tr>
<tr>
<td>Carpet</td>
<td>127 Lux</td>
<td>89 Lux</td>
<td>Level 1</td>
</tr>
</tbody>
</table>

Table 3: Light levels for the tests run indoors

<table>
<thead>
<tr>
<th>Surface</th>
<th>Light intensity before the test run</th>
<th>Light intensity after the test run</th>
<th>Light level</th>
</tr>
</thead>
<tbody>
<tr>
<td>White sheet</td>
<td>894 Lux</td>
<td>750 Lux</td>
<td>Level 2</td>
</tr>
<tr>
<td>Paisley duvet</td>
<td>571 Lux</td>
<td>556 Lux</td>
<td>Level 2</td>
</tr>
<tr>
<td>Carpet</td>
<td>482 Lux</td>
<td>571 Lux</td>
<td>Level 2</td>
</tr>
</tbody>
</table>

Table 4: Light levels for the tests run indoors
<table>
<thead>
<tr>
<th>Surface</th>
<th>Light intensity before the test run</th>
<th>Light intensity after the test run</th>
<th>Light level</th>
</tr>
</thead>
<tbody>
<tr>
<td>White sheet</td>
<td>77,5k Lux</td>
<td>78,5k Lux</td>
<td>Level 3</td>
</tr>
<tr>
<td>Paisley duvet</td>
<td>77,5k Lux</td>
<td>78,5k Lux</td>
<td>Level 3</td>
</tr>
<tr>
<td>Grass</td>
<td>77,5k Lux</td>
<td>78,5k Lux</td>
<td>Level 3</td>
</tr>
</tbody>
</table>

Table 5: Light levels for the tests run outdoors

To do the testing various sorts of equipment were used. Here follows a list of them:
- Samsung Galaxy S7 edge [55]
- Linocell Tripod for the phone to be able to have the phone stationary in different angles
- Light meter MT-906 which measures in lux
- Protractor for measuring angles
- Measuring tape for measuring the distance of the phone to surfaces

5.2.4 Analysis of collected data

Criteria to interpret the findings within the study emerged while the literature study was conducted. Which resulted in three different data elements: (i) time to detect a plane, (ii) size of a plane and (iii) amount of detected feature points. See section 4.8.4, Analysis of collected data. Number of feature points detected is one way of interpreting and analyze the results of the study, however whether an object can be placed out on the scene or not by a user, is one of the main things which defines how well-functioning the application really is. We rank it as better if the size of the plane is bigger since that means that ARCore see more of the surface as an available plane. A bigger plane also means that a user has more options when placing 3D objects on the scene. Therefore the criteria regarding planes are of great importance in evaluating ARCore. An analysis of the results received within the procedure of the data collection can be found in chapter seven, Data analysis. Whereby statistical analyzes were performed to determine whether the mean values calculated from the collected data were of significant difference, and thereby meaningful to draw conclusions from.

5.2.5 Presentation and conclusion of outcome

In order to make a comparison between the results which were received, the raw data was compiled and is presented by graphs in chapter 6, Result. Raw data is not included in the report, however could be available if the study or analysis is to be repeated or if there is a need to assure the quality of it. The results are presented based on the criteria that were intended to be analyzed. Conclusions regarding the case study conducted and its outcome were made, things to have in mind if a similar study was to be conducted again were documented, these are presented in chapter eight, Conclusions.

5.3 Software development

The phase of developing the application, the test cases and executing the tests lasted for four weeks. The implementation phase began with a start-up meeting with the supervisor of the project at Slagkryssaren. Following work tools were introduced and set up: (i) Bitbucket, a web-based version control repository hosting service for source code and development projects which make use of Mercurial or Git [56] and (ii) Trello, a web-based project management application used in order to easily track ongoing projects [57].

Initially, a board was created through Trello, with a product backlog consisting of the primary tasks/issues that needed to be done in order to progress within the development work. Functionality desired by the supervisor, the requirements, were defined and put into the backlog. Guidance of things that should be done before the actual coding was started was given by the supervisor. The way of working applied during the project was also determined at the start-up meeting. One task should be in progress at a time, which meant that it was worked on together until it was completed, thereafter the next task could be addressed. Scheduled meetings were to be conducted daily and weekly with the supervisor: (i) daily stand-up meetings which would last for approximately five minutes, similar to Daily Scrum(described in section 3.2.1) and (ii) weekly follow-up meetings which would last for approximately 45 minutes, similar to Sprint Review(see section 3.2.1). Internal weekly meetings were also conducted in order to plan the work which should be done during the following week, also referred to as iteration. A goal was set up for each iteration, according
Git was used as version control system during the process of developing the application, managed through Bitbucket. A mock-up of the application was created in order to facilitate the development of it, but also for the purpose of clarifying the functionality needed to enable the evaluation of the application and the data collection.

5.3.1 Application

HelloAR.java is a sample project provided by Google, including code and helper-classes, protected by the Apache 2.0 license. The sample project allows the user of the application to place out Android figures on surfaces within the scene, given that a plane has been detected. A figure is placed out on the scene by tapping a detected plane and the position of the object remains even if the device is moved into another view. However, there is a limitation of how many figures that can be present on the scene simultaneously, when this limit is reached the first object which was placed out on the scene is removed. Thus, this project was borrowed and used as foundation during the study and development work.

5.3.2 Added features

The first thing that was done was trying out the sample project in order to become familiar with its features. When that was done new features could be added. A splash screen was added which runs the first time the application is started. The splash screen shows the instructions in how to use the application, the benchmarking methods and a button to continue to ARmode. In ARmode, a method was created to collect data from the application. Each second (by default), this method collected the current number of detected points, the X-size and the Z-size of a possibly detected plane and the time for which it took to detect the first plane. These values were collected as separate java objects with a timestamp connected to it. The collection was done for a set amount of time (20 seconds by default). When the collection was done these java objects were added to a .json file. The .json file was then mailed to a specific mail address and collected there. A menu was added to the ARmode screen from which one could access three menu choices. These menu choices include returning to the splash screen to read the instructions again, one function to send an email of current benchmark, and lastly a button which takes one to the settings screen. At the settings screen one can choose whether or not benchmode should be activated (i.e. the collection of data), number of total anchors available for placement. One could also choose for how long the benchmarks should run and at what interval points and planes should be collected. The benchmark times are 10, 20 and 30 seconds and the interval of collecting is collect every other second, collect every second and collect two times each second.

A java program was created that could parse multiple .json files. The program calculates the average of detected feature points, the average size of X and Z planes and the average time it takes to detect the first plane. These means are calculated for every second by considering all executions of the same test. A link to the source code of the project can be found in Appendix A.
6 Result

Here follows the various results that have been collected during the tests, presented as graphs. Section 6.1 will go through the comparison of the three tested light levels. Section 6.2 will go through the comparison of the three chosen angles. Section 6.3 will go through the comparison of the four chosen surfaces and how they worked. All of these tests have been performed five times each and an average of those five runs has been calculated and is presented here in the form of graphs. All y-axes within following graphs indicate an average per second. Corresponding graphs which illustrate the development of data second to second are found in the appendix. All graphs found in this chapter and in the appendix are generated from our data and created by us.

6.1 Comparing light levels

Below follows the results of the testing with different light levels. The testing was done with three different light levels and two different types of surfaces. The phone was moved back and forth during these tests.

![Figure 8: Paisley](image)

![Figure 9: Paisley XZ product](image)

Figure 8 shows the result of the motion test for the paisley duvet with a comparison between the three light levels. Light level 3 produced the largest number of points in light level 3 than during the other two.
light levels. Light level 1 seem to have produced more points than light level 2.

Figure 9 shows data from the same test as the previous figure but with the size of the plane as the metric instead of number of points. Light level 3 yielded a larger plane than the two other light levels, as with the previous figure. However, when studying light level 1 and 2, it can be seen that in the matter of light level 2 a larger plane was produced, unlike the number of detected points presented in figure 8, where light level 2 ended up with minimum number of feature points.

Figure 10 shows the result of the motion test for the white sheet with a comparison between the three light levels. Light level 3 yielded more feature points than during the other two light levels, with a big margin. Light level 2 produced slightly less points than light level 1.

Figure 11 shows the result from the same test as the previous figure but with the size of a plane as a metric. The results of the white bed sheet as surface follow the same trend as the paisley duvet cover. Light level 2 produces a larger plane according to figure 11 than light level 1 even though a larger number of feature points were detected during light level 1. What also can be seen in figure 11 is that light level 3 yielded the largest plane of them all.
6.2 Comparing angles

Below follows the results of the testing with the chosen angles. The tests were done with three different surfaces, three different angles in two different light levels with the phone in a stand. During these tests no planes were detected by ARCore. So the only data included here is the average number of points detected per second, whereby data was collected during a 20 second interval.

![Figure 12](image1.png)

**Figure 12:** Angle comparison with carpet as surface at light level 1

![Figure 13](image2.png)

**Figure 13:** Angle comparison with carpet as surface at light level 2

Figure 12 illustrates how the three different angles produced feature points against the carpet used as surface at light level 1. There is almost no difference at all when looking at number of detected feature points at 65 degrees in comparison with 90 degrees. Figure 13, where the same tests were performed as the previous figure but under the circumstances of light level 2. The results are almost reversed when compared to the previous figure. The angle of 90 degrees barely produced any points.
The graph presented in figure 14 shows how the three different angles produced feature points as the paisley duvet was used as surface at light level 1. In this case the angle of 65 degrees resulted in the greatest number of detected feature points. This was also the case when the same test was conducted under the circumstances of light level 2, see figure 15, the angle of 65 degrees resulted in the greatest number of detected feature points.
The result of placing the device in the three different angles when the white bed sheet was used as surface at light level 1 is shown in figure 16. The difference between the angles is not significant, however when the test was performed at light level 2 the angle of 45 degrees clearly produces the largest number of detected feature points, which can be seen in figure 17.
Figure 18 shows the average of how many points the different angles produced during all these tests, namely light level 1, light level 2, carpet, paisley duvet and white bed sheet. There is no significant difference in number of detected feature points between the angle of 45 degrees and 65 degrees, the angle of 90 degrees produced the minimum number of points. As mentioned earlier, no planes were detected during any of these tests.

### 6.3 Comparing surfaces

Here follows the results of our testing of surfaces. These tests were done with four different surfaces in three different light levels. The device were moved in a repeated motion over the surface at the same height and speed during these tests. Graphs with number of detected points and size of plane will be present in these graphs.

![Graph of average number of detected points during all stationary angle tests](image)

**Figure 18:** The average number of points per angle collected from all angle tests

**Figure 19:** Surface comparison - Light level 1 - Moving - Feature Points
Figure 19 shows the result of the motion test for light level 1 with a comparison between the three different indoor surfaces. The paisley duvet produced by a big margin the largest number of points, the carpet surface produced the second largest number of points and the white sheet the least amount of points.

Figure 20 shows how large planes were produced during the same test shown in the previous figure. Interestingly enough, the carpet surface produced the largest average plane which contrasts the results from the previous figure where the paisley duvet produced more points overall. The paisley duvet produced the second largest average plane and the white sheet the smallest average plane by a large margin.

Figure 21: Surface comparison - Light level 2 - Moving - Feature Points
Figure 21 shows the result of the motion test for light level 2 with a comparison between the three different indoor surfaces. The metric used is average number of detected points. As figure 19 also shows, the paisley duvet produced the largest number of points during this light level too. The same goes for the carpet surface and the white sheet which produced in the similar way as in figure 19 with the difference that the carpet is closer in detected points to the paisley duvet rather than the sheet.

Figure 22 shows how large planes were produced during the same test shown in the previous figure. This figure stands in contrast to what was shown in figure 20. Here the paisley duvet produced the largest average plane, the sheet produced the second largest average plane and the carpet the smallest average plane. The carpet produced more points in this test than the sheet but produced a smaller plane than it.

Figure 23: Surface comparison - Light level 3 - Moving - Feature Points
Figure 23 shows the result of the motion test for light level 3 with a comparison between the two indoor surfaces from before and a new outdoor surface. As seen in the previous tests, the paisley duvet produced the largest number of points, the white sheet produced the second largest number of points and the grass surface the least amount of points. What differs from previous tests here is that the other surfaces produce almost equally as many points as the paisley duvet.

Figure 24 shows how large planes were produced during the same test shown in the previous figure. The grass surface produced the largest average plane, the white sheet produced the second largest average plane and the paisley duvet the smallest average plane by a large margin. The paisley duvet produced the most points during this test but the smallest plane whereas the grass surface produced the least amount of points but the largest average plane.

Figure 25 shows the average time for detection of a plane for the indoor testing of surfaces. This graph has the average for both light level 1 and light level 2. The paisley and the carpet had very similar detection times. The white sheet was the slowest by almost double the time as the other two.
Figure 26: Average time for detection of plane - Outdoors

Figure 26 shows the average time for detection of a plane for the outdoor testing of surfaces, the overall average for light level 3. This differs from the previous graph in the way that the sheet and the grass surface produce a plane at a similar rate. The paisley is the one surface that almost doubles the time it takes compared to the other two.

Figure 27: Average time for detection of plane - Total

Figure 27 shows the total average of both the previous graphs. The white sheet produced a plane at the overall slowest pace. The carpet produced a plane at the quickest rate compared to the other. The performance of the paisley duvet and the grass are quite close to each other.
7 Data analysis

This chapter presents the statistical analysis done on the acquired data. Section 7.1 presents the ANOVA tests done on the benchmarked data. Section 7.2 presents the post-hoc testing done on those ANOVA tests that showed a significant statistical difference to determine the source of that difference.

7.1 ANOVA test

ANOVA tests were performed on the results presented in section 6 to determine if there was any significant statistical difference in them. The tests were done to each corresponding figure's original data, comparing three groups to each other. In these tests it is checked whether the resulting F is larger than F critical. If F is larger than F critical then it can be concluded that there is a significant statistical difference between the groups. If F is smaller than F critical then it can be concluded that there is no significant statistical difference. The significance level was set to 0.05.
<table>
<thead>
<tr>
<th>Figure number</th>
<th>Description</th>
<th>F</th>
<th>F Critical</th>
<th>F&gt;F Critical ? Yes : No</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>Paisley; Light Levels; Moving Motion; Detected Points</td>
<td>8.064366</td>
<td>3.158843</td>
<td>Yes</td>
</tr>
<tr>
<td>9</td>
<td>Paisley; Light Levels; Moving Motion; Size of Plane</td>
<td>2.102653</td>
<td>3.158843</td>
<td>No</td>
</tr>
<tr>
<td>10</td>
<td>Sheet; Light Levels; Moving Motion; Detected Points</td>
<td>46.03991</td>
<td>3.158843</td>
<td>Yes</td>
</tr>
<tr>
<td>11</td>
<td>Sheet; Light Levels; Moving Motion; Size of Plane</td>
<td>23.3151</td>
<td>3.158843</td>
<td>Yes</td>
</tr>
<tr>
<td>12</td>
<td>Carpet; Angles; Light Level 1; Stationary; Detected Points</td>
<td>11.03692</td>
<td>3.158843</td>
<td>Yes</td>
</tr>
<tr>
<td>13</td>
<td>Carpet; Angles; Light Level 2; Stationary; Detected Points</td>
<td>33.70169</td>
<td>3.158843</td>
<td>Yes</td>
</tr>
<tr>
<td>14</td>
<td>Paisley; Angles; Light Level 1; Stationary; Detected Points</td>
<td>35.40018</td>
<td>3.158843</td>
<td>Yes</td>
</tr>
<tr>
<td>15</td>
<td>Paisley; Angles; Light Level 2; Stationary; Detected Points</td>
<td>78.16242</td>
<td>3.158843</td>
<td>Yes</td>
</tr>
<tr>
<td>16</td>
<td>Sheet; Angles; Light Level 1; Stationary; Detected Points</td>
<td>0.486975</td>
<td>3.158843</td>
<td>No</td>
</tr>
<tr>
<td>17</td>
<td>Sheet; Angles; Light Level 2; Stationary; Detected Points</td>
<td>100.6426</td>
<td>3.158843</td>
<td>Yes</td>
</tr>
<tr>
<td>18</td>
<td>Average of All; Angles; Stationary; Detected Points</td>
<td>15.57424</td>
<td>3.021012</td>
<td>Yes</td>
</tr>
<tr>
<td>19</td>
<td>Indoors; Surfaces; Light Level 1; Moving Motion; Detected Points</td>
<td>19.0758</td>
<td>3.158843</td>
<td>Yes</td>
</tr>
<tr>
<td>20</td>
<td>Indoors; Surfaces; Light Level 1; Moving Motion; Size of Plane</td>
<td>6.467767</td>
<td>3.158843</td>
<td>Yes</td>
</tr>
<tr>
<td>21</td>
<td>Indoors; Surfaces; Light Level 2; Moving Motion; Detected Points</td>
<td>12.4876</td>
<td>3.158843</td>
<td>Yes</td>
</tr>
<tr>
<td>22</td>
<td>Indoors; Surfaces; Light Level 2; Moving Motion; Size of Plane</td>
<td>0.104591</td>
<td>3.158843</td>
<td>No</td>
</tr>
<tr>
<td>23</td>
<td>Outdoors; Surfaces; Light Level 3; Moving Motion; Detected Points</td>
<td>2.669734</td>
<td>3.158843</td>
<td>No</td>
</tr>
<tr>
<td>24</td>
<td>Outdoors; Surfaces; Light Level 3; Moving Motion; Size of Plane</td>
<td>4.709205</td>
<td>3.158843</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Table 6: Analysis of variance

Table 6 showed that out of the 16 tests done 12 of those showed a significant statistical difference. Only four of them did not show a significant statistical difference. The 12 tests that showed a difference needs to be post-hoc tested to determine the source of the difference.

### 7.2 Post-hoc test

In those tests that resulted in a significant statistical difference in table 6, i.e. those tests where F was greater than F critical, post-hoc tests were performed to determine between which groups there were a significant statistical difference. Three post-hoc tests were performed per each ANOVA test. Every ANOVA test consisted of three groups each which resulted in three post-hoc tests, one test between group
1 and 2, one test between group 1 and 3 and one test between group 2 and 3. These post-hoc tests were performed via t-tests between two groups were a Bonferroni correction was used. If P is less than the Bonferroni correction it is assumed that there is a significant statistical difference between the two groups. If not then it is assumed that there is no significant statistical difference between the two groups. To get the Bonferroni correction one takes the used significance level divided by the number of tests. With a significance level set to 0,05 and with three different Post-hoc tests the Bonferroni correction we receive equals to 0,05 / 3 = 0,167.

<table>
<thead>
<tr>
<th>Figure number</th>
<th>Light Level</th>
<th>Light Level</th>
<th>P(T&lt;=t) two-tail</th>
<th>Bonferroni Correction</th>
<th>P&lt;Bonferroni Correction ?</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>Level 1</td>
<td>Level 2</td>
<td>0,276265887</td>
<td>0,167</td>
<td>No</td>
</tr>
<tr>
<td>8</td>
<td>Level 1</td>
<td>Level 3</td>
<td>0,010715057</td>
<td>0,167</td>
<td>Yes</td>
</tr>
<tr>
<td>8</td>
<td>Level 2</td>
<td>Level 3</td>
<td>0,000748388</td>
<td>0,167</td>
<td>Yes</td>
</tr>
<tr>
<td>10</td>
<td>Level 1</td>
<td>Level 2</td>
<td>0,573127056</td>
<td>0,167</td>
<td>No</td>
</tr>
<tr>
<td>10</td>
<td>Level 1</td>
<td>Level 3</td>
<td>2,11391E-08</td>
<td>0,167</td>
<td>Yes</td>
</tr>
<tr>
<td>10</td>
<td>Level 2</td>
<td>Level 3</td>
<td>6,9641E-09</td>
<td>0,167</td>
<td>Yes</td>
</tr>
<tr>
<td>11</td>
<td>Level 1</td>
<td>Level 2</td>
<td>0,021072129</td>
<td>0,167</td>
<td>No</td>
</tr>
<tr>
<td>11</td>
<td>Level 1</td>
<td>Level 3</td>
<td>1,23351E-07</td>
<td>0,167</td>
<td>Yes</td>
</tr>
<tr>
<td>11</td>
<td>Level 2</td>
<td>Level 3</td>
<td>0,000226685</td>
<td>0,167</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Table 7: Light Levels

In table 7 it is seen that post-hoc tests were done for figure 8, 10 and 11. Figure 9 showed in the ANOVA test that there were no significant statistic difference between the groups. **Figure 8** (the light level test with the paisley duvet, feature points) shows no significant statistic difference between light level 1 and light level 2. Is is shown to be a significant statistic difference between light level 2 and 3 and between light level 1 and 3. **Figure 10** (the light level test with the white sheet, feature points) shows no significant statistical difference between light level 1 and light level 2. A significant statistic difference is shown between light level 2 and 3 and between light level 1 and 3. **Figure 11** (the light level test with the white sheet, size of plane) shows no significant statistic difference between light level 1 and light level 2. Between light level 2 and 3 and between light level 1 and 3 it is shown to be a significant statistic difference though.
<table>
<thead>
<tr>
<th>Figure number</th>
<th>Angle</th>
<th>Angle</th>
<th>P(T\leq=t) two-tail</th>
<th>Bonferroni Correction</th>
<th>P&lt; Bonferroni Correction?</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>45 degrees</td>
<td>65 degrees</td>
<td>0.000150433</td>
<td>0.167</td>
<td>Yes</td>
</tr>
<tr>
<td>12</td>
<td>45 degrees</td>
<td>90 degrees</td>
<td>5.24573E-05</td>
<td>0.167</td>
<td>Yes</td>
</tr>
<tr>
<td>12</td>
<td>65 degrees</td>
<td>90 degrees</td>
<td>0.560415772</td>
<td>0.167</td>
<td>No</td>
</tr>
<tr>
<td>13</td>
<td>45 degrees</td>
<td>65 degrees</td>
<td>0.110118361</td>
<td>0.167</td>
<td>No</td>
</tr>
<tr>
<td>13</td>
<td>45 degrees</td>
<td>90 degrees</td>
<td>5.87857E-10</td>
<td>0.167</td>
<td>Yes</td>
</tr>
<tr>
<td>13</td>
<td>65 degrees</td>
<td>90 degrees</td>
<td>3.51669E-09</td>
<td>0.167</td>
<td>No</td>
</tr>
<tr>
<td>14</td>
<td>45 degrees</td>
<td>65 degrees</td>
<td>2.31127E-10</td>
<td>0.167</td>
<td>Yes</td>
</tr>
<tr>
<td>14</td>
<td>45 degrees</td>
<td>90 degrees</td>
<td>1.27757E-07</td>
<td>0.167</td>
<td>Yes</td>
</tr>
<tr>
<td>14</td>
<td>65 degrees</td>
<td>90 degrees</td>
<td>0.012627123</td>
<td>0.167</td>
<td>Yes</td>
</tr>
<tr>
<td>15</td>
<td>45 degrees</td>
<td>65 degrees</td>
<td>6.00366E-11</td>
<td>0.167</td>
<td>Yes</td>
</tr>
<tr>
<td>15</td>
<td>45 degrees</td>
<td>90 degrees</td>
<td>0.016189694</td>
<td>0.167</td>
<td>Yes</td>
</tr>
<tr>
<td>15</td>
<td>65 degrees</td>
<td>90 degrees</td>
<td>1.31715E-10</td>
<td>0.167</td>
<td>Yes</td>
</tr>
<tr>
<td>17</td>
<td>45 degrees</td>
<td>65 degrees</td>
<td>5.98225E-11</td>
<td>0.167</td>
<td>Yes</td>
</tr>
<tr>
<td>17</td>
<td>45 degrees</td>
<td>90 degrees</td>
<td>2.06088E-13</td>
<td>0.167</td>
<td>Yes</td>
</tr>
<tr>
<td>17</td>
<td>65 degrees</td>
<td>90 degrees</td>
<td>2.50791E-11</td>
<td>0.167</td>
<td>Yes</td>
</tr>
<tr>
<td>18</td>
<td>45 degrees</td>
<td>65 degrees</td>
<td>0.640112757</td>
<td>0.167</td>
<td>No</td>
</tr>
<tr>
<td>18</td>
<td>45 degrees</td>
<td>90 degrees</td>
<td>5.18253E-06</td>
<td>0.167</td>
<td>Yes</td>
</tr>
<tr>
<td>18</td>
<td>65 degrees</td>
<td>90 degrees</td>
<td>1.0164E-09</td>
<td>0.167</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Table 8: Angles

In table 8 it is seen that post-hoc tests were done for figure 12, 13, 14, 15 and 17. Figure 16 showed in the ANOVA test that there were no significant statistic difference between the groups of that test. **Figure 12** (the angle test with the carpet surface, light level 1, feature points) shows a significant statistic difference between the 45 degree angle and the 65 degree angle, and also between the 45 degree angle and the 90 degree angle. The 65 degree angle and the 90 degree shows no significant statistic difference. **Figure 13** (the angle test with the carpet surface, light level 2, feature point) shows that there is no significant statistic difference between the 45 degree angle and the 65 degree angle or between the 65 degree angle and the 90 degree angle. When comparing the 45 degree angle and the 90 degree angle it is shown to be a significant statistic difference. **Figure 14** (the angle test with the paisley surface, light level 1, feature point) shows a significant statistic difference on all of the three angles. **Figure 15** (the angle test with the paisley surface, light level 2, feature point) also shows a significant statistic difference on all of the three angles. **Figure 17** (the angle test with the white sheet surface, light level 2, feature point) it is also shown that there is a significant statistic difference on all of the three angles. **Figure 18** (the overall angle comparison with all surfaces and all light levels compared, feature point) it is no significant statistic difference between the 45 degree angle and the 65 degree angle. Between both the 45 degree angle and the 90 degree angle, and also between the 65 degree angle and the 90 degree angle it is a significant statistic difference.
In table 9 it is seen that post-hoc tests were done for figure 19, 20, 21 and 24. Figure 22 and 23 showed in the ANOVA test that there were no significant statistic difference between the groups in those tests so no post-hoc tests were needed for them. Figure 19 (the surface test with light level 1, feature points) shows a significant statistic difference between both the white sheet and the paisley duvet and also between the paisley duvet and the carpet surface. When comparing the white sheet and the carpet surface the test no significant statistic difference. Figure 20 (the surface test with light level 1, size of plane) shows no significant statistic difference between either the white sheet and the paisley duvet or between the paisley duvet and the carpet surface. The white sheet and the carpet surface shows a significant statistic difference when compared. Figure 21 (the surface test with light level 2, feature points) shows a significant statistic difference between both the white sheet and the paisley duvet and also between the white sheet and the carpet surface. When the paisley duvet and the carpet surface is compared it is shown to be no significant statistic difference. Figure 24 (the surface test with light level 3, size of plane) shows a significant statistic difference between both the white sheet and the paisley duvet and also between the paisley duvet and the grass surface. Between the white sheet and the grass surface there is no significant statistic difference.
8 Conclusions

This chapter presents the conclusions regarding the work which was conducted during the thesis. Below follows a reflection regarding the privacy when it comes to using augmented reality applications, a discussion concerning the choice of methods which were applied during the thesis. Section 8.2 discusses the choice of methods that were used during this work. Section 8.3 presents an analysis of the results. Overall conclusions regarding the work are discussed in section 8.4. Delimitations together with suggestions of improvement are discussed in section 8.5. Finally, future work is discussed.

8.1 Ethical aspect

The technology of augmented reality is quite impressive and could definitely have some social benefits, however there may also be a downside of it, such as ethical issues. Considering what has happened lately with the incidents regarding Facebook [58] and their way of handling data privacy, one should be aware of that augmented reality could entail big ethical concerns.

Augmented reality is used, inter alia, to entertain, manipulate and inform end users, while their data is being collected and utilized by companies who develop augmented reality applications. One area of concern is how this information is used and utilized and of which. Wherein a strategy of developers and companies to persuade end users to use their products is to customize them according to the behavior of the users. This is done by collecting as much data as possible regarding the actions of the end users, which becomes a problem if intrusion on a person's privacy is the way to success. Thus, it is a matter of how the privacy of end users is protected and how the process of getting permission to unveil personal information is conducted [59, 60].

8.2 Choice of methods

Regarding the choice of methods and whether it has affected the outcome of the thesis in some way is quite hard to distinguish. However, conducting a case study and developing software with an agile approach, are both examples of working in a flexible manner. The circumstances which the data collection procedure was conducted, and the tests that were aimed to be performed, nearly demanded a flexible approach. Wherein the data collection could be adapted to observed phenomenon as the procedure of it progressed.

Initially it was not obvious which tests that should be performed and evaluated according to the evaluation criteria. It was uncertain which kind of data that should be collected and saved by the application. Since the product backlog is allowed to change and does not need to be all set before developing the software, and the fact that a functioning product with some of the desired features can be delivered after just one iteration, makes it grateful to work according to the agile approach and make progress.

On the other hand, according to the waterfall model and its linear approach, you can only progress from one stage of the development process to another if it is considered to be fully complete. This way of working could have been beneficial to some extent, if all requirements had been decided from the beginning, perhaps there had been more time to explore ARCore's feature detection.
8.3 Analysis of results

Figure 28: The total average feature points per second for light test

Figure 29: The total average plane size per second for light test

Figure 28 shows the total average feature points per second for each light level and figure 29 shows the average plane size. It can be seen in these figures that light level 3 in overall performed the best, both in terms of detected points and size of plane. For the paisley duvet we had the best average number of detected points and size of plane at light level 3. The same with the white sheet which also performed better at light level 3, both in terms of detected points and size of plane. According to the ANOVA test and following post-hoc test done in section 7 there is a significant statistical difference between light level 3 and light level 1 or 2 in regards of points for both the paisley duvet and the white sheet. The white sheet also had a significant statistical difference at light level 3 compared to both light level 1 and 2 in regards of the size of the plane. The paisley duvet did not have that difference in regards to the size of the plane. When comparing light levels 1 and 2 between each other it is not as clear cut which light level gives better results. According to the ANOVA test and post-hoc test there is no significant statistical difference between light level 1 and 2, both regarding the detected points and the size of the plane. For the paisley duvet, light level 1 produced a higher number of detected points than light level 2. A larger plane was produced during light level 2 which is something that is ranked better than the number of points. For the white sheet, light level 1 produced a higher number of detected points than light level 2. Light level 2 did produce a larger average plane than light level 1. Overall, these results show that when doing a
**moving motion** with your device a **brighter light level** will be more in favor for ARCore to detect planes however if the difference between the light levels is small it does not seem to matter as much.

In section 6.2 where the results of the **angle tests** were compared it is shown that the results are varied. At light level 2 the paisley surface only detected points during the 65 degree angle whereas the white sheet mostly produced its majority of detected points during the 45 degree angle. Figure 18 shows the total average during all these tests. The 45 degree angle and the 65 degree angle were the angles that produced the most feature points. The 90 degree angle produced the least which feels expected in the way that there is less showing of the surface in that angle. The 45 degree angle produced at an average a little more than the 65 degree angle but it is hard to determine if that is circumstantial or a good result. According to the ANOVA tests done in section 7, all but one test (the angle comparison with the sheet at light level 1) showed to have significant statistical differences between them. In the post-hoc tests only 4 out of the 18 tests did not show any statistical differences between them. For the overall comparison seen in figure 18 it is shown in the ANOVA and post-hoc tests that there is no significant statistical difference between the 45 degree and the 65 degree angle. Between the 45 degree and the 90 degree angle and between the 65 and the 90 degree angle there is a difference statistically.

What all these angle tests have in common is that they did not produce any planes at all. What can be concluded from this is that having your device stationary will be less in favor for ARCore to detect planes when compared to the tests done in 6.1 and 6.3 where the device was used in a **moving motion**.

*Figure 30: The total average feature points per second for surface test*
Figure 31: The total average plane size per second for surface test

Figure 30 shows the total average of the surface tests done at light level 1 and 2 and how many feature points were produced by each surface at an average. Figure 31 shows the total average plane size for the different surfaces. It is seen in figure 30 that the paisley duvet produced the most feature points, the carpet surface the second most and the sheet the least. In figure 31 it is seen that the carpet surface produced the largest average plane, the paisley duvet the second largest plane and the sheet the smallest plane. It was expected that the white sheet would produce the least amount since it could prove harder to detect changes in a blank white sheet in comparison to something with texture or features. For light level 3, which is shown in figure 23 and 24, the paisley duvet produced the largest amount of points but produced the smallest plane. The grass surface produced the smallest amount of detected points but the largest plane. The white sheet produced a larger plane than the paisley duvet this time. This could be a result of potential creases getting even more exposure at a light level this bright. The ANOVA and poc-hosts tests show that there were no statistical deviations in 2 of the 6 tests. At light level 1 (figure 19) it is shown that the carpet and sheet had no statistical difference between them, but they both had when compared to the paisley surface in regards to points. In regards to the plane (figure 20) size it shows that the only difference in statistics was between the sheet and the carpet. At light level 2 (figure 21) there is no statistical difference between the paisley and the carpet, but when comparing the sheet and the paisley, and the sheet and the carpet there is one, when comparing points. At light level 3 which looks at plane size (figure 24) it is shown to be a significant statistical difference between the sheet and the paisley, and the paisley and the grass surface. Between the grass and the sheet there is no significant statistical difference.

What this concludes is that at light level 1 the paisley duvet produced more points than the others, but it is not certain if the carpet produced the largest plane statistically. At light level 2 it can be concluded that both the paisley duvet and the carpet produced more points than the white sheet but it is not certain if paisley produced more points than the carpet. The plane sizes at this light level does not give any of the surfaces an edge over the other by looking at the statistics. At light level 3 it can not be concluded which of the surfaces produced the most points. It can be concluded that both the grass surface and the white sheet produced a larger plane than the paisley duvet. It cannot be said if the carpet produced a larger plane than the white sheet or not.

By looking at figure 27 which shows the average times to detect planes for the different surfaces, we can see that the carpet and the grass were the fastest to detect a plane. The paisley produced a plane at a slightly slower pace and the white sheet produced a plane at the slowest pace.

Overall, these results show that it is advantageous for ARCore to be pointed at surfaces that produce textures in some repeatable way as with the grass and the carpet. The white sheet does not provide textures in the same way and could be a disadvantageous surface to use. Paisley duvet provides lots of patterns and colors, but this could prove to be bad for ARCore since it is more information to process, and less repeatability as seen with the grass and carpet.

It is hard drawing concrete conclusions to these tests, more testing and more test cases would be preferable. Overall, from these tests it seems that there is a preference for ARCore to have a light environment when used, for best results an outdoor light environment even. There seems to be an overall preference for ARCore that the surface it is pointed at features textures with some sort of repeatability, like grass or a carpet. Surfaces without textures are less of a preference for ARCore. Using a moving motion with your device rather than having it stationary seems to be preferable for ARCore too since it detected planes that way but did not when it was stationary. This could be because of the limited depth perception of a mobile phone which has only one perspective and that depth perception requires two perspectives (such as the human eyes)[61]. This could require the device to be moved to gain new perspectives. ARCore seemed to be benefited by using it at a 45 or 65 degree angle rather than using a 90 degree angle.

8.4 Overall conclusions

Augmented Reality has already been implemented within areas such as health care, retail and education, however there are far more potential fields of application for the technology. What could threaten its success is whether the technology is accepted by the users, both in ethical aspects and in terms of usability, whether the quality of the technology meets the expectations of the users or not, this thesis has investigated how far the development of the technology behind ARCore has come.

One of the purposes of the report was to provide building blocks for other researchers to continue evaluating ARCore and its feature detection. This work and paper has provided some building blocks which future researchers could make use of. How the tests were done could be used in future works, even the tests that were not performed could be put to use. How the data was collected from the application could
also be made of use for these testing purposes with same, or altered, evaluation criteria. The same sort of tests could be performed with the different light levels and different surfaces to see if it would yield the same results.

The other purpose was to give an indication to companies and developers of the current state of ARCore and its feature detection it uses. This purpose would need more research to come to conclusions in some aspects. What can be concluded from the testing is that at this state ARCore is not suitable with a stationary device and that it performs better in a lighter environment. This information could prove useful for companies that are developing applications for Android and would want to utilize ARCore.

The main goal with the work was to map how well-functioning ARCore's feature detection is and whether it functions differently depending on the varying environmental circumstances. That goal was reached in a way that it could be determined that ARCore's feature detection behaved differently during certain environmental circumstances. As shown in the results in section 6, ARCore differs in its behavior when the light level, surface and angle differs in any way.

In ARCore’s current state, the possible applications we see are in the area of entertainment. Games and applications like previously mentioned Poké mon GO where you can get augmented reality overlays onto your display. The focus of the applications would benefit from having the activity take place outside in combination with walking. Overall it would be best suited to have it analyze environments with a high light level while moving the device. An application handing out simple step-by-step instructions could be made possible via ARCore, the limitation in that application scenario is that the room can not be too dimly lit or having device still. ARCore would overall be less suitable where the purpose of the application is to have it stationary in some way. We think it might be a bit early for ARCore to be used in applications or programs where it is critical that the application is working accurately, for example if the application is to be used within health care or construction or something along those lines. A more refined platform would probably be needed with specific solutions to the possible problems that application might be used for.

8.5 Delimitations

There are some factors that have been observed that may have affected the outcome of the study in a disadvantageous manner, which could be great to have in mind in the future if the case study was to be conducted again. Following are some reflections regarding the circumstances which the tests were performed.

The room in which the tests were carried out consisted of, as mentioned before, a window. The light emitted from the window made the general light level of the room constantly change, even though the general light level was measured right before and after each test run, the measurements could perhaps be a bit misleading. The white sheet which was aimed to correspond to a white surface without texture, was hard to get completely outstretched. Shadows occurred within the scene which may have generated feature points that would not have been detected if a white surface without texture would have been used. The limited light conditions which could be achieved indoors resulted in that some tests needed to be performed outdoors. When the tests were run outdoors, textures were created within the scene due to the wind, which may have generated feature points that would not have been detected otherwise.

Thus, it would have been preferable to carry out the tests in an environment where the circumstances could be controlled. One idea would have been to test the same criteria for each surface. For instance perform a stationary test with an angle of 45 degrees for each surface, before moving on to the next angle in order to increase the chance of same light conditions, and therefore the same environmental circumstances. However, there was no access to a lab to perform the tests in, the best was made out of the situation.

Doing these same tests that have been done with an ARKit program could also prove useful in a way to compare how ARCore stands against Apple’s software development kit. That would have required more time and would change the purpose and goals for the project a little. It would also have proven useful to do these tests with other augmented reality frameworks intended for Android development.

8.6 Future work

Whether ARCore’s feature detection would be researched again it would be preferable to have an object present at the scene. That way it could be compared how well ARCore detects objects in certain conditions and that object could be used as a reference. Future works could also test more parts of the ARCore framework such as battery usage and performance tests. These tests are unrelated to ARCore’s feature
detection but related to the overall package that is ARCore and the information that could be collected from these tests could prove useful for consumers and corporations interested in developing through ARCore.
References


Appendices

Appendix A provides a link to the project on GitHub, Appendix B presents a range of graphs which illustrates the development of data second to second. The data which was collected during the tests, a mean value from five tests for a 20 seconds duration is presented.

A Appendix

To see the source code for the application follow this link: https://github.com/starkelove/ARCore-kex

B Appendix

Figure 32: SHEET - Light levels compared. Average from 5 tests for a 20 second duration. Feature points.

Figure 33: SHEET - Light levels compared. Average from 5 tests for a 20 second duration. Size of plane.
**Figure 34:** PAISLEY - Light levels compared. Average from 5 tests for a 20 second duration. Feature points.

**Figure 35:** PAISLEY - Light levels compared. Average from 5 tests for a 20 second duration. Size of plane.
Figure 36: CARPET - Angle comparison, light level 1. Average from 5 tests for a 20 second duration. Feature points.

Figure 37: CARPET - Angle comparison, light level 2. Average from 5 tests for a 20 second duration. Feature points.
Figure 38: SHEET - Angle comparison, light level 1. Average from 5 tests for a 20 second duration. Feature points.

Figure 39: SHEET - Angle comparison, light level 2. Average from 5 tests for a 20 second duration. Feature points.
Figure 40: PAISLEY - Angle comparison, light level 1. Average from 5 tests for a 20 second duration. Feature points.

Figure 41: PAISLEY - Angle comparison, light level 2. Average from 5 tests for a 20 second duration. Feature points.
**Figure 42:** Surface comparison, light level 1. Average from 5 tests for a 20 second duration. Feature points.

**Figure 43:** Surface comparison, light level 1. Average from 5 tests for a 20 second duration. Size of plane.
**Figure 44:** Surface comparison, light level 2. Average from 5 tests for a 20 second duration. Feature points.

**Figure 45:** Surface comparison, light level 2. Average from 5 tests for a 20 second duration. Size of plane.

**Figure 46:** Surface comparison, light level 3. Average from 5 tests for a 20 second duration. Feature points.
**Figure 47:** Surface comparison, light level 3. Average from 5 tests for a 20 second duration. Size of plane.