2019

Developing Contextual Approximation within Augmented Reality: An Exploration of The Effectiveness of Personalised Augmented Reality Technology

MSC ADVANCED COMPUTER SCIENCE: C1875958

DANIEL LAGES

Table of Contents

Abstract

Augmented reality is a prominent computing paradigm. Used within a range of scenarios, the presentation of graphical software overlays allows the technology to provide detailed information concerning the current environment. As explored throughout this project. many attempts have been made in order to optimize the paradigm for different circumstances in which it may provide a greater level of utility.

Such optimisations are made in conjunction with the implementation of corresponding technologies such as optical character recognition and location gathering. This project explores the use of such components and how they can be adopted in conjunction with augmented reality in order to derive context.

Through the development and evaluation of an application, a determination is made concerning the effectiveness and limitations of contextually aware augmented reality technologies. This project concludes that, although providing utility in a narrow use case, broader scenarios in which such a technology may operate require a large amount of infrastructure in order to provide relevant information to the user.

Table of Figures

List of Tables

1. Introduction

Augmented reality is an expanding area within personal computing. The paradigm has opened avenues concerning personalised data exploration, navigation and gaming. As a result of the range of use cases, it is regarded by many as a powerful computing paradigm with many applications yet to be fully explored.

Differing from virtual reality, augmented reality (AR) concerns the overlaying of software components on the physical environment in relation to the perspective of the user. Often achieved through the use of a camera module within a computational architecture, the software components serve to interact with or modify the physical environment such that relevant information relating to a particular scene can be provided.

Various computational paradigms are often adopted to coordinate an output based upon a rage of input vectors. These metrics, within the context of AR, can include but are not limited to location and the regularity of behaviour. It is therefore apparent that the effectiveness of the AR paradigm is reliant on the capabilities of the computational methodologies adopted by the system.

Such methodologies may be used in order to develop contextual awareness within AR applications. Contextual awareness describes the analysis of the current environment or situation, in which the result of such computation is the presentation of a state that provides information relevant to the user.

Implemented in a number of environments, contextual awareness can serve to improve efficiency and performance. For example, the use of AR in the medical sector has proven beneficial during surgeries, providing a visual overlay of vital structures that may be hidden beneath an insertion point. Contextual awareness is important within this scenario, allowing for any unnecessary information to be discarded from the user interface, ensuring focus and visibility (Akbarinasaji and Homayounvala 2017).

Extended to more personalised experiences, augmented reality can operate within a different set of constraints. Early consumer facing developments within this sector have approached this challenge but have been unsuccessful; in regard to mass consumer adoption. This is as a result of a range of factors, such as hardware limitations. More fundamentally however, the lack of relevant use cases has often been cited as a core reason for the lack of exposure; alongside other issues such as ergonomics and security. (Hilken et al. 2017)

All AR technologies serve as an extension of the current tools used by consumers every day, enacting a further abstraction or removal from the obtuse nature of many of the devices in use at the time of writing. This is achieved through the construction of environments that serve to merge physical and digital interactions. To abridge, the paradigm allows for the generation of a canvas that may act to facilitate real time and contextually aware information based upon various computational techniques, executing as a result of recognisable behavioural constraints.

1.1 Project Statement

This project will develop and demonstrate an understanding of how various computational methodologies can be combined with the Augmented Reality paradigm in order to implement contextual awareness, displaying relevant data within a given circumstance through the analysis of different parameters.

A specific use case will be implemented in order to determine the effectiveness of the employed techniques and how they can be optimised in conjunction with AR technologies. This use case will depict the user’s journey through a train station. Chosen because of the various AR components that will be tested, this scenario will involve the analysis of signage and other parameters in order to derive context, ensuring the program developed remains suited to the topic domain.

1.2 Objectives

- To explore the Augmented Reality paradigm and develop an understanding of its many use cases.
- To explore the implementation methodologies that can be used in order to develop an Augmented Reality application.
- To investigate the different computational techniques that can be implemented, in order to develop contextual awareness through the analysis of provided datapoints and constraints
- Demonstrate an understanding of the different implementation technologies that can be used in order to incorporate such computational techniques.
- Develop a specific test case that describes the effectiveness of the Augmented Reality paradigm.
- Implement an augmented reality application that applies differing computational techniques in order to operate within the context of the devised use case
- Evaluate the effectiveness of the augmented reality paradigm when adopting computational techniques in order to achieve contextual awareness.

2. Background Material

2.1 Augmented Reality

Augmented Reality describes the overlay of digital elements onto the physical surroundings, through the use of a computational device. The paradigm allows for an extension of the natural world through the generation of a digital interface, highlighting new interactions and facilitating behaviours within the environment, to the benefit of the user (Bonnard et al. 2013)

The ability of AR technologies to improve user experience through the addition of new software oriented graphical interactions is present within many unique contexts. For example, as stated by (Schwarz, Fastenmeier 2017) AR can be used in order to improve the awareness and safety of a user within a vehicle, highlighting certain aspects of a given journey such as road warnings and speed limits. The use of this technology within this context, highlights its versatility and useful nature.

This is further exemplified in differing industrial sectors, in which AR has become a pivotal technology in the workplace. The technology can be used in a range of occupations, in which the superposition of user interface elements on top of a relevant scenario serves to aid workflow. For example, tasks involving the maintenance of machinery could adopt the graphical overlays facilitated by the AR paradigm, in order to improve user perceptions and efficiency (Akbarinasaji and Homayounvala 2017).

As stated by (Nex et al. 2015), the many features that the AR paradigm provides allows it to serve as a prominent tool within industries such as archaeology. Permitting the user to interact with an artefact and explore archaeological landmarks from a distance and perhaps in more detail. In addition to a greater level of convenience, the use of the paradigm may also limit the negative effects of handling an object of a significant age.

The medical sector may also benefit from the implementation of AR technology. This is exemplified by (Katic et al. 2013) who states that the use of AR in scenarios such as laparoscopic surgery may serve advantageous in highlighting important areas in relation to an incision point. For example, the correct use of the technology may ensure that arteries are avoided, decreasing the likelihood of risk.

(Katic et al. 2013) goes on to describe the constraints faced during the construction of AR technologies today. Describing the paradigm as incapable of displaying data in a contextually efficient manner, often leading to the convolution of data and, thus, obstructing the user's field of view. As a result, displaying contextually aware data will serve to improve the effectiveness of the paradigm. This is a view supported by (Ong et al. 2014), describing the potential for AR to aid the completion of a range of tasks that are regular in nature; suggesting that AR could offer greater assistance through a conceptual understanding of the procedure or action at hand.

The AR paradigm is also growing to a greater level of prominence within commerce, providing a pivotal movement towards new customer experiences. As stated by (Hilken et al 2017) the use of AR technology allows for consumer to more greatly explore the products offered by an online company.

The sufficient implementation of this paradigm provides a visualisation method (detailing how a product may look within the home of a user for example). This is a technique employed by a range of commercial companies such as IKEA, using the technology in order to showcase differing items of furniture as if within the home of the consumer (IKEA 2019)**.** Ultimately, this paradigm allows for the greater removal of technological steps from the purchasing experience. Through the adoption of such methods, a progressive abstraction away from technology is achieved such that its implementation is hidden (Hilken et al. 2017).

2.3 Personalised Augmented Reality

Beyond industry, the AR paradigm provides many use cases within the realm of personal and consumer technology. It is widely believed that AR will bring a profound change to the human experience. As a result, it is estimated that the implementation of the technology will position both the technology industry and social practices within a new era. This is analogous to the impact that derived from the introduction of the smart phone (Kipper and Rampolla, 2013)

Vast amounts of research and development expenses have been implemented by consumer facing companies in order to develop devices and features, such that AR is adopted as a useful paradigm (Google 2019).

Despite such investments in the growth of the technology, the paradigm has seen restricted success due to many technical limitations. Object recognition, for example, is a vital method employed within the analysis of a physical environment. Conversely, (Huang 2013) suggests that the technological advances made in order to achieve object recognition have yet to obtain the standard of success required. The resource goes on to describe the need for the paradigm to achieve the recognition and computation of different images within a range of unique instances. An AR application may, for example, operate with more noise and less accuracy than another (as a result of the optimisation towards a particular scenario). Producing different and inconsistent levels of utility, at the detriment of the user.

AR applications or 'browsers', as detailed by (Tsai and Huang, 2018), must overcome of a range of factors. The delivery of information is of vital importance. For example, strides must be made in order to ensure that the layout and format of information is considered such that the display does not overwhelm the user. Prompting the completion of multiple tasks, or the analysis of a large amount of information at a given time, will lessen the effectiveness of the paradigm. This reasoning is perhaps more relevant in accordance with the realities and restrictions of augmented reality glasses, an implementation that requires more regular interaction on a smaller screen (Tsai and Huang, 2018).

Regardless of the limitations of the paradigm, a number of consumer-facing applications have risen to prominence**.** (Kim et al. 2016) describes the wide adoption of AR games. Platforms such as Pokémon GO has attracted a large number of users (Niantic 2019). The game, which encourages exercise through the obtainment of digital creatures upon the physical landscape, has also served to improve user wellbeing (Kim et al. 2016).

Further benefits have been realized in conjunction with the paradigm. For example, through the use of the camera component and image recognition capabilities, assistance can be provided to the visually impaired. A use case that is analyzed by (Huang 2019) who state that the development of such technology can be used to magnify areas and text of interest, such as signage. A system was developed that serves to highlight text through optical character recognition technology. This project again emphasises the need to carefully tune the information displayed, determining that the presence of a vast amount of signage can lead to an "obtrusive" user interface that obstructs the vision of the user (Huang 2019).

A range of other use cases have also been investigated concerning AR technology. For example, areas such as navigation have become increasingly popular within the consumer adoption of AR. Within such applications, graphical overlays serve to display arrows denoting a required direction for the user to reach a desired destination. A further example is the use of AR within museums, providing guided tours through a serious of graphical presentations, initiated based on the recognition of a scene or user location (Carmaniginiani et al. 2011).

Referring to the creation of an augmented reality application that is non-obtrusive, (Singh and Singh, 2013) state that the user should be able to determine the difference between software components and reality so to reduce user error. Further challenges are outlined within this resource, describing the need to filter information based upon the intent of the user.

Different methods can be adopted in order to filter information. For example, intelligent algorithms can be utilised in order to determine the granularity of which data is displayed. Such algorithms are explored by (Choi et al 2011). This resource describes the importance of the hierarchical structuring of data when visualising the environment. Through the use of machine learning techniques such as the Nearest Neighbour algorithm, the presentation of data can be more greatly refined in order to benefit the user (Choi et al. 2011).

This challenge again highlights the need for contextual awareness within the paradigm, allowing the user to benefit from the abstraction of the technology and provide information only when relevant, so not to clutter the interface and limit cognitive load. The implementation of differing computational techniques, such as optical character recognition and location services, can be used to provide contextual awareness.

Throughout the current implementations of the AR paradigm, a range of other challenges have been observed. The adoption of AR has faced many restrictions such as rendering limitations and tracking time latency. However, these challenges are largely as a result of hardware limitations. Through evolving the software capabilities of AR platforms, greater functionality can be achieved. There is a clear need to develop the AR paradigm beyond locational awareness, in order to understand other forms of context that may serve to improve the functionality that it can provide (Gruber 2017).

2.4 Augmented Reality Implementation

A range of advances have been made in order to allow developers to implement AR functionality. Particular strides in the mobile sector has allowed for developers to design mobile applications through the use of a range of Software Development Kits (SDKs). The implementation of new methodologies alongside the development of more capable mobile devices has led to a dramatic growth within the sector. This is exemplified with the inception of Apple's AR-Kit and connected modules (Wang 2018).

The functionality of an AR application consists of the completion of a range of steps, often processed upon a single computational device. Initially, the surrounding environment is captured. This is achieved through the use of a camera component or the computation of location. The environment information is then processed in order to determine any elements of the data that can be augmented, this is often achieved through analysis of an existing database or the initiation of a specified function. The results of such computation serve to activate the presentation of software elements, in line with the functionality of the app (Singh and Singh 2013).

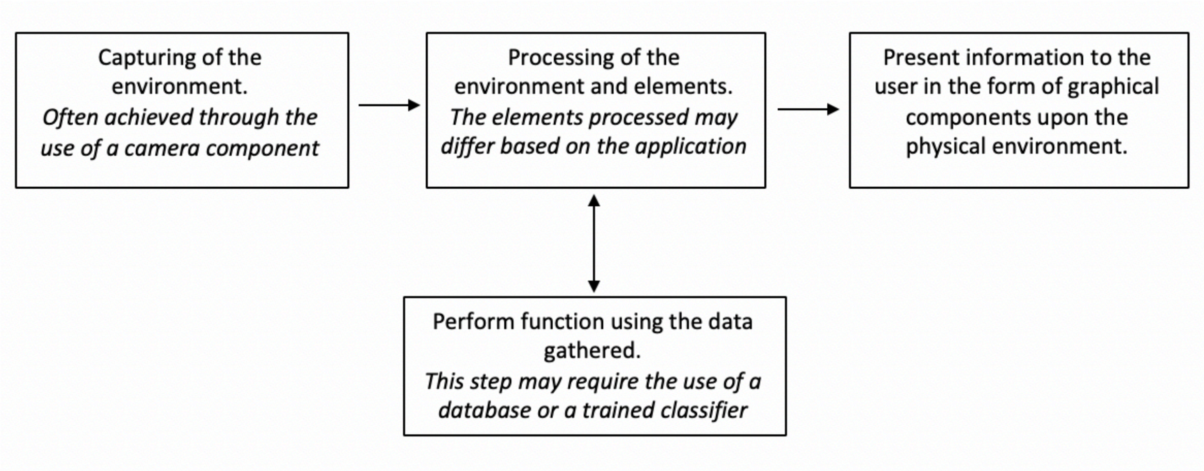

Capturing of the environment.
Often achieved through the use of a camera component
Processing of the environment and elements.
The elements processed may differ based on the application
Present information to the user in the form of graphical components upon the physical environment.
Perform function using the data gathered.
This step may require the use of a database or a trained classifier

Figure 1: The Steps of computation performed by an AR Application

To determine the contextual constraints on which the technology can compute, a range of methodologies have been explored. For example, the detection of a certain object within a given scene may serve as a trigger for a precise function. The detection of a specific identification system, for example. provides a greater awareness of the position of an object within a given space. This is a methodology explored by (Olson 2011) in which visual fiducial systems known as April Tags were developed, facilitating motion capture and the superposition of information upon real-world objects.

The use of April Tags provides a range of benefits over more simplistic visual fiducial systems such as QR tags. Requiring less definition and clarity in order to be registered by the camera component (Olson 2011). The use of such a system can allow for the greater analysis of a 3D space and are often employed to gather information about the angle of a given object, providing further contextual information about a given scene.

Location is perhaps one of the most prominent methods used by AR applications, displaying information based upon a request for directions. The ability to show points of interest has been well investigated and implemented thoroughly (Tsai and Huang, 2018). Such data can be gathered through the use of APIs that harness Global Positioning System (GPS) technology. The Technology used within the Core Location interface provided by Apple, facilitates this, for example (Apple 2019).

In order to build upon the data gathered through collecting the location of the user, text spotting can be used to gather further information about a scene. Through determining the text associated with a point of interest that is local to the user, further parameters can be added to the computation performed. As a result, more reliable data can be retrieved, and more contextual information provided.

This method is explored by (Gubbi and Amruter 2018), which describes the development of a convolutional neural network model that can be trained in order to recognise text. The model developed uses “1,800 open fonts” in order to ensure reliability in a range of different environmental scenarios. The resource also describes the need for different lighting and contrast conditions that must be trained upon.

This article describes the adoption of the sliding window method, a technique also explored by (Elaalyani 2017) in which a “classifier window’ is used in order to detect text within a scene. The trained classifier will then operate upon the highlighted area in order to classify the text (Gubbi and Amruter 2018).

In order to mitigate the downfalls of this technique, the current image may be downsized and the relationship between individual characters learned, reducing the number of false positive classifications. Regardless of the methodology adopted in order to achieved text spotting and OCR, the processing time must be optimal in order to be suitable for the AR paradigm, providing users with the appropriate information efficiently.

Analysing user behaviour is a further method that can be adopted in order to develop contextual approximation. A range of states can be defined in order to categorize how consumers may interact with a virtual environment.

This is described by (Tsai and Huang, 2018) in which certain behaviors are categorized so to determine when the user is actively looking for points of interest for example. In this scenario, the information displayed to a user in this state will be minimized. In states that require further detail however, such as when the user is focusing on a point of interest, the information provided may be more comprehensive and relevant to the information point. It is clear that the determination of user behavior can serve as an important metric for the implementation of the paradigm.

Perhaps the biggest barrier to the implementation of AR on a large, consumer facing scale, is that of hardware engineering. For example, determining how software elements can interact with the environment can be difficult in outdoor areas. As described by (Behzadan et al. 2008), unlike indoor environments (in which sensors are able to map the parameters of the space) spaces with wider scale are harder to track. This is as a result of a range of considerations, such as changing points of interest and the correlation of data in relation to the position of the user.

At a fundamental level, AR technologies concern the creation of a virtual environment, serving as a canvas for the viewing and analysis of data. The virtual environment is derived from one of many techniques; as outlined by (Kovlev, A. 2007) when exploring the development of 'personal-use' displays. For example, one well established method for the generation of a virtualised environment results from the execution of computational tasks, such as rendering based upon the analysis of symbols upon a 3D object.

Such engineering challenges may require more advanced hardware progressions that are not ready for mass market adoption at the time of writing. Advancements in camera technologies to recognize points of interest, GPS for mapping user locations and battery component for providing power over long periods are all important developments to be made in this sector Behzadan et al. 2008). The addition of supportive devices that may serve to support the computation of wearable AR devices may serve as a prominent avenue in order to mitigate these restrictions.

2.5 Contextual Awareness

The introduction of contextually aware functionality within the AR paradigm can serve to improve the utility it provides, analysing information relevant to the current environment in order to apply the software such that it is pertinent to the existing scenario (Ong, S.K. 2014).

Many sources of data can be used in order to derive context. (Mehra 2012), state that data derived from the location of the user, user preferences and the functional history of the system can all be coordinated in order to generate context. Modelling such information can be achieved in a number of ways and allows for logical rules to be generated, such that unique functions can be performed.

In order to map functionality to a given context, a range of methodologies can be used. As explored by (Grubert et al. 2017) a taxonomy can be developed in order to outline the different contextual factors that must be operated upon. Such parameters are categorised as “contextual sources” that may include the detection of a range of actions or scenarios. Contingently based upon on the functionality provided by the AR program, human behaviour may be considered a key factor.

Through the analysis of location data and the positioning of the user within the current environment, the information provided can serve to meet the specific action of the user. As denoted by (Sassamn 2014) the computation of user position by an AR system can be achieved on a small scale without the use of a GPS sensor, instead adopting methodologies such as object recognition. This is supported by (Kayas et al. 2015) who determines that the computation of a location state allows for the description of “the spatial characteristics of an entity”.

Current activity can also be employed as a parameter. This is particularly applicable within maintenance applications, as explored by (Ong, S.K. 2014) Based upon the recognition of a scheduled or deduced task, the context–aware system is able to access information relating to the current action. The computation of this task may allow for certain and relevant information to be gathered and, as a result, the filtering of information to suit the needs of the user. As outlined by (Katic et al. 2013), this is a key challenge that must be overcome in the development of pragmatic AR applications.

One can determine a range of practices for the development of contextually Aware AR systems through a broader analysis of software methodologies. (Kayas et al. 2015) investigates how context can be recognised in order to determine the access rights to a given resource, in the interest of “security and privacy” within occupational confines.

Within this resource, an ontology is designed to define the factors used and determine context and model responses. “Context Entities” are determined, providing key information about the resource or functionality being requested. Such information may include the owner of the resource, the resource history and location. Using the information stored, a resource request can be verified through the application of simple logic-based rules based upon the structure of the ontology; applying a knowledge of context and the relationships between entities in order to grant or dismiss permissions (Kayas et al. 2015).

Applied to AR, such context information and logic-based rules can be applied. For example, regarding a ‘context entity’ constraint such as time and the behaviour of the user, certain functionality can be presented to or hidden from the user based upon the understanding of these constraints.

3. The Problem

The completion of the background material has highlighted the prominence of the AR application and how it can be adopted in order to provide new utility and a greater integration of technology with human behaviour. Despite the benefits provided by the paradigm, there is an evident demand to expand on the functionality delivered, in order to allow for the recognition of context.

The inclusion of such computation will provide many benefits, overcoming a range of problems specified within the background material section of the report. The ability to provide, limit and filter information based upon a range of input parameters has been explored through the investigation of different research and development projects.

3.1 Competing Products

A range of research projects have been carried out concerning the development of the AR paradigm, the effects it may have and the utility that it can provide. (Grubert and Zollman, 2017) investigate the implementation of a range of techniques that may be used in order to develop contextually aware AR systems.

For example, different taxonomies and other methods for contextual approximation can be implemented, providing a mechanism for determining when to activate certain functionality. The taxonomy developed by this resource discusses the use of 'context targets' that are analysed by the system in order to determine what information to display. Other sources of context may also be considered. For example, 'human factors' such as the movement of a user can also be employed to depict intention (Grubert, 2017).

The development of contextually aware computation is also explored by (Lukowicz 2011) who state that the computational analysis of high level processes, as opposed to the lower level tasks that they consist of, allows for a more abstract outcome to be characterised and used as a datapoint during the training of a context aware system.

In support of this methodology, the resource also describes the challenges of developing contextually aware systems. As contextual parameters may change based upon the user of a device, individual user-focused training is highlighted as method that may be utilised in order to identify the behaviour attributed to the user.

Despite the positive effect that this methodology may provide, gathering user feedback about the effectiveness of the computation system may be a convoluted process. For example, new interface paradigms may need to be designed, permitting responses from the user at given stages. The implementation of this functionality must also manage any new behaviours and one-time events, so not to predict behaviour based upon noisy data (Lukowicz 2011).

(Siddiqui et al. 2013) also investigate how context can be analysed within an AR system, with the aim of determining the significant objects in the environment. The article describes how the correct implementation of scene understanding technologies can lead to advances in how the AR system can function within a given scenario.

This methodology can be adopted in order to overcome many challenges faced by current AR technologies. For example, (Katic et al. 2013) outlines the need to filter certain information when guiding laparoscopic surgeries. Contextual awareness can be applied in order provide location and structure data based on prior experiences. A precise estimate of the anatomy can be delivered such that only the location of relevant arteries and organs are detailed as software overlays.

(Siddiqui et al. 2013) also cite tasks such as object recognition and navigation as key applications of such technology, if implemented effectively. In order to recognise individual object of a scene, feature extraction is used, taking in account the most predictable elements of a typical environment. From here, the less dominant objects can be extracted in order to derive context.

There are many products and applications have been developed for use within industrial and consumer facing sectors. Once a promising consumer product. Google Glass has now been aimed solely towards industry, with many citing the hardware design and use of a camera a prominent reason for its demise in consumer facing segments (Sullivan 2017).

The use of contextually aware AR has also been explored by (Ong et al. 2015) who suggest that an understanding of context "can improve the usability of AR systems". The paper illustrates how software methodologies can be created in order to model context. For example, the development of ontology is determined an important step in the construction of logic-based responses to the current scenario, analysing parameters such as the present activities, location and time.

The ability of the AR paradigm to provide assistance to those with impairments has been explored by (Sassaman 2014). The article, states that the capacity of AR technologies to "track the position of the user" and display corresponding annotations provides a range of advantages. The resource also discusses the challenges faced when fetching information, stating that when developed for multiple large-scale applications, data should be derived using crowed-sourcing methods in order to analyse recognisable entities at scale.

Many of the successes that have been observed concerning the AR paradigm have occurred in the form of mobile applications. Applications such as Pokémon Go (an app that uses location data in order to derive context) serve as prominent indications of the effectiveness of the paradigm (Niantic 2019).

The use of location data in order to derive context is a theme explored by (McCall et al. 2011) who investigate how the parameter can be used to provide greater functionality. Although this resource concerned the analysis of AR games, it provides insight into how location is an important consideration in providing useful features.

Despite this, many of the AR applications developed are created only to meet a single use scenario and are unable to analyse the contextual information within the environment when meeting a diverse set of objectives. As a result, many implementations of the paradigm are unable to provide proactive data when desired, instead requiring the user to evoke the user interface with an appropriate action.

3.2 Response

In response to the challenges faced concerning the development and use of contextually aware AR technology, this project will develop a program that adopts a range of methods and technologies. The approach used in order to develop the program will encompass many of the techniques implemented by similar projects, with the aim of furthering and combining the results obtained through the combination such methods. The implementation of such techniques will be explored, analysing various API's and libraries in order to achieve OCR, gather location parameters and recognise environments.

An ontology will also be devised in line with (Ong et al. 2015) detailing how various parameters gathered will be used in-order to derive context. It is vital to the development of the system that all operations are specified in coordination with a single user, allowing for personalised context to be derived. In order to store this data, a database implementation will be explored and implemented.

To improve focus and ensure a good level of functionality given a set of time and resource constraints, a scenario will be determined. This scenario will act to provide an environment that can be used as a benchmark when developing the technology and evaluate the resulting outcome. Although the technology will be created so to meet the required functionality, as set out by the scenario, the project will aim to utilise a range of methodologies that can be use on a wider and more universal scale.

4. Approach and Methodology

Through the development of an application, this project will explore how the augmented reality paradigm can be extended with the addition of contextually aware computation. A range of features will be developed in order to achieve a context aware system and provide relevant information within the devised scenario.

It has been determined that the project will adopt the Waterfall methodology in order to provide structure during the development of the program and composition of the report. This methodology implements a sequential approach to the completion of tasks.

It has been decided that this method aligns with the consecutive nature in which the software components are implemented within the project. For example, each element (such as a OCR) required the gathering of background research in order to ensure that the best approach was taken. The waterfall methodology also allows for the overlapping of tasks as an integral component, providing a more responsive way to overcome challenges and plan for any contingency periods relating to time delays.

It was determined, in line with suggestions provided by the supervisor, that a plan should be devised outlining the expected start and completion of tasks. The plan was completed in accordance with the waterfall methodology, providing a list of targets concerning the completion of certain tasks. Prior research was also carried out, determining the greater accuracy, the expected length of time that should be outlined for the implementation of a given technologies.

Having liaised on a weekly basis, the supervisor provided key insight into a range of methodologies and approaches. At each stage of the project, feedback was sought and retrieved so to ensure that any developments were made, and a range of avenues explored.

The application will be created through the use of the Software Development Kit (SDK) provided by Apple, alongside the Swift programming language (Apple, 2017). It has been determined that the use of these components provide a well-documented platform to build upon, using a range of programming packages and APIs. The combination of these technologies will allow for the implementation of the functionality required; in order meet the criteria set out by the test case.

The application developed will be created for devices running the iOS operating system developed by Apple. A decision made in response to the portability such devices offer, representative of the way in which AR is optimally used (Apple, 2019).

4.1 The Scenario

The application will be developed in order to meet the standards of a devised scenario. The scenario covers a range of different functions in order to ensure an array of information is gathered, improving the understanding of the environment and any intentions that the user may have. Providing the developed system with the most optimal circumstances to implement the design functionality.

The scenario depicts a train station, with various signs and platforms serving as 'information points'. With a pre-determined knowledge of the destination that the user would like to reach, the aim of the AR application is to provide relevant platform and route information in the form of software-based alerts.

Within this scenario, the AR application will employ different methods that correspond with the findings made in the research section of this report. Location data will be gathered and utilised in order to determine preliminary information about the environment, such as to trigger the remaining functions.

Functionality will also be incorporated in order to read and analyse tags, such as the implementation of reference images (in line with the implementation of April Tags) as discussed by (Olson 2011). The adoption of this technology will allow for the swift recognition of certain information points such as train times and platform signs. Upon the recognition of such information points, text spotting and Optical Character Recognition (OCR) techniques will be used. This will facilitate the identification of text and the resulting obtainment of route and platform information. At each step, the application will present graphical information in relation to the data gathered, providing assistance to the user. Such information will consist of train times, platform information and directions.

The interface displayed will be designed and presented in line with research findings, in which the amount of information must be minimal and should not obstruct the view of the user. To achieve this, a rule-based system will be developed, ensuring appropriate and contextually suitable representations of information. For example, the functionality provided by the application will only be initiated upon entry to the environment specified by the scenario (the recognition of a train sign). The process specified by the scenario is outlined in figure 2.

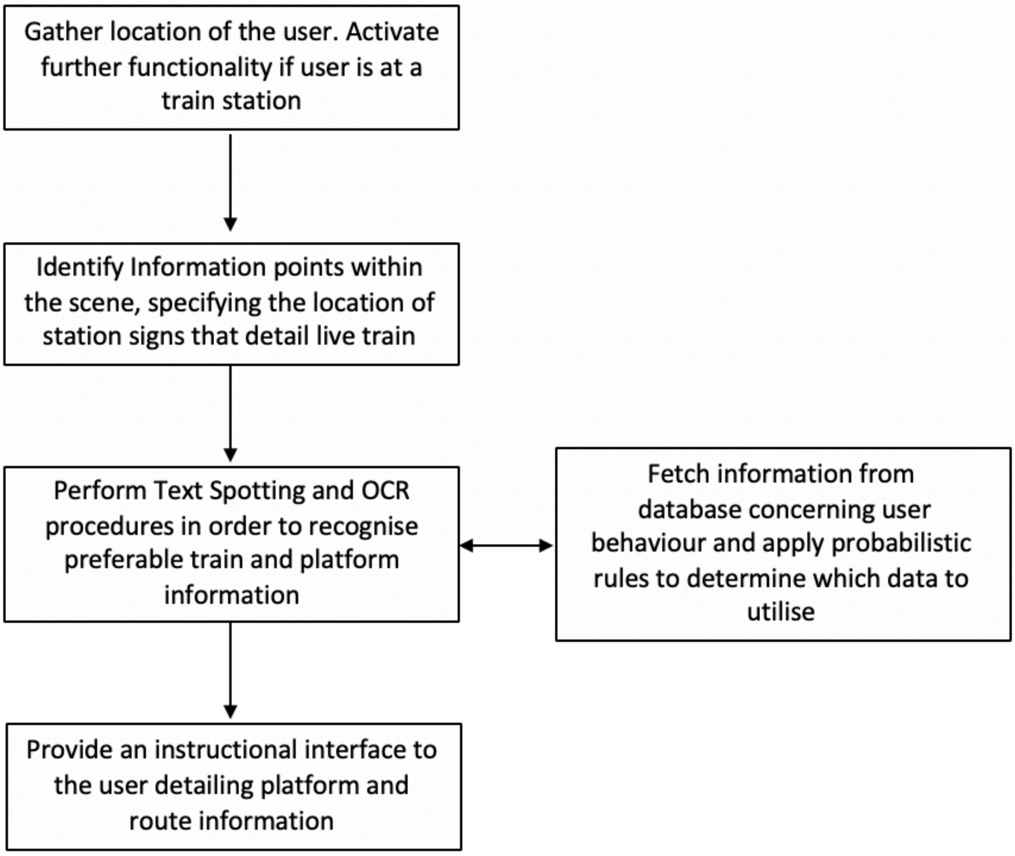
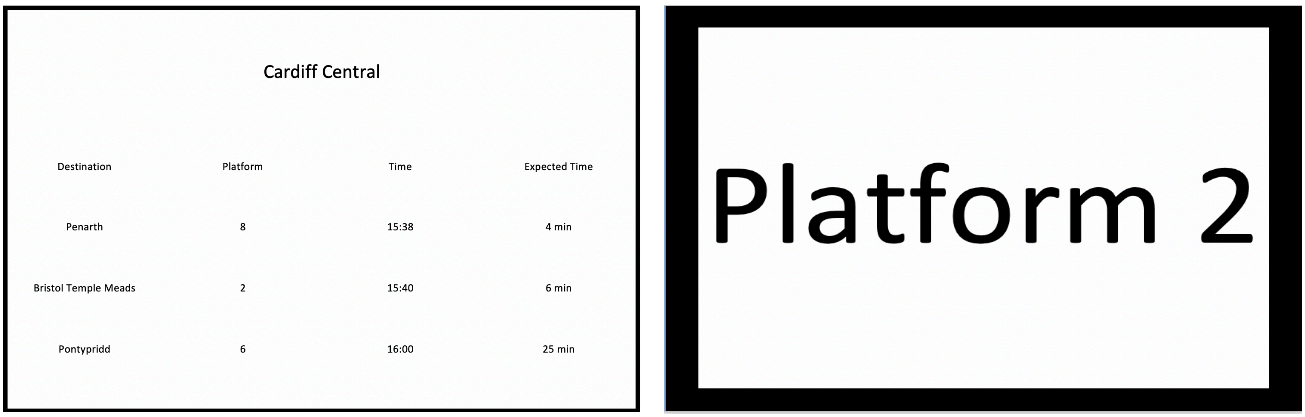

Gather location of the user. Activate further functionality if user is at a train station
Identify Information points within the scene, specifying the location of station signs that detail live train
Perform Text Spotting and OCR procedures in order to recognise preferable train and platform information
Fetch information from database concerning user behaviour and apply probabilistic rules to determine which data to utilise
Provide an instructional interface to the user detailing platform and route information

Figure 2: The steps to be completed by the developed AR application

The 'information points' will be designed in line with the conventions described by typical train station signs. Despite this, colours and font will be adjusted so to provide the optimal conditions for the implementation of environment recognition and OCR. As a result, each sign will be generated using a white background and black font. Figure 3 represents the information points developed in accordance with the scenario.



Cardiff Central

Destination	Platform	Time	Expected Time
Penarth	8	15:38	4 min
Bristol Temple Meads	2	15:40	6 min
Pontypridd	6	16:00	25 min

Platform 2

Figure 3: Initial Information Point Design

4.2 Location Data

As specified by (Carmaniginiani et al. 2011) location data is a key metric, often adopted by AR applications. It is used in many cases to trigger certain functionality, and as a result, allows for a primitive form of context to be recognised. Many APIs have been developed that allow the location of a currently active device to be retrieved through the use of technologies such as GPS, Wi-Fi and Bluetooth.

In line with the adoption of the iOS platform and the Swift programming language, the Core Location API will be used, as provided by Apple. 'Core Location' allows for the management of different location services at various levels of granularity. The identification of such locations behaviours could allow for greater predictive capabilities, improving contextual awareness (Apple 2019).

Alongside the implementation of location data, the application will also gather information concerning the current time. Each event record utilises a 'time description' variable, providing an abstract metric that allows for the greater determination of context. Using this metric, the system will gain an understanding of the most relevant train times at a given interval.

4.3 Environment Recognition

As supported by (Olson 2011), the recognition of identifiable tags or data can allow for the analysis of the environment, gathering data concerning the position of the user and the location of significant objects local to the device. The ARkit SDK will be used in order to analyse recognisable information points and activate related functionality. For example, upon the recognition of an information point such as a platform sign, a graphical overlay can be presented, providing clear directional instructions to the user (Apple, 2019).

4.4 Optical Character Recognition

Upon the detection of text within a scene, a number of frameworks can be implemented in order to extend this functionality and achieve OCR. This project will apply the Tesseract OCR engine in order to recognise the characters defined within the pre-determined areas of text (Google, 2019). In order to integrate the OCR engine for use within the ARKit application, the 'SwifityTesseract' module will be used, providing a method for merging tesseract with the Swift programming language (SwiftyTesseract, 2019)

Following the implementation of OCR and the retrieval of a result, a function will be applied in order to parse certain aspects of the result, thus, ensuring that all relevant information is stored for future computation. For example, the 'Platform' variable will be allocated a value based upon the recognition of the associated platform values within an information point. This value will then be implemented in order to direct the user to the correct area of the train station.

4.5 Rule-Based Contextual Awareness

In line with the scope afforded by the test-case, an ontology can be established, defining the relationship between differing contextual parameters and activation of predefined functionality. It has been determined, throughout the literature review that analysing and understanding the behaviour of the user is vital for the understanding of context within a given scenario. As supported by (Tsai and Huang, 2018), the identification of different user states can assist in determining the most suitable information to display. Figure 4 outlines the relationship between the contextual parameters analysed by the system in order to determine context.

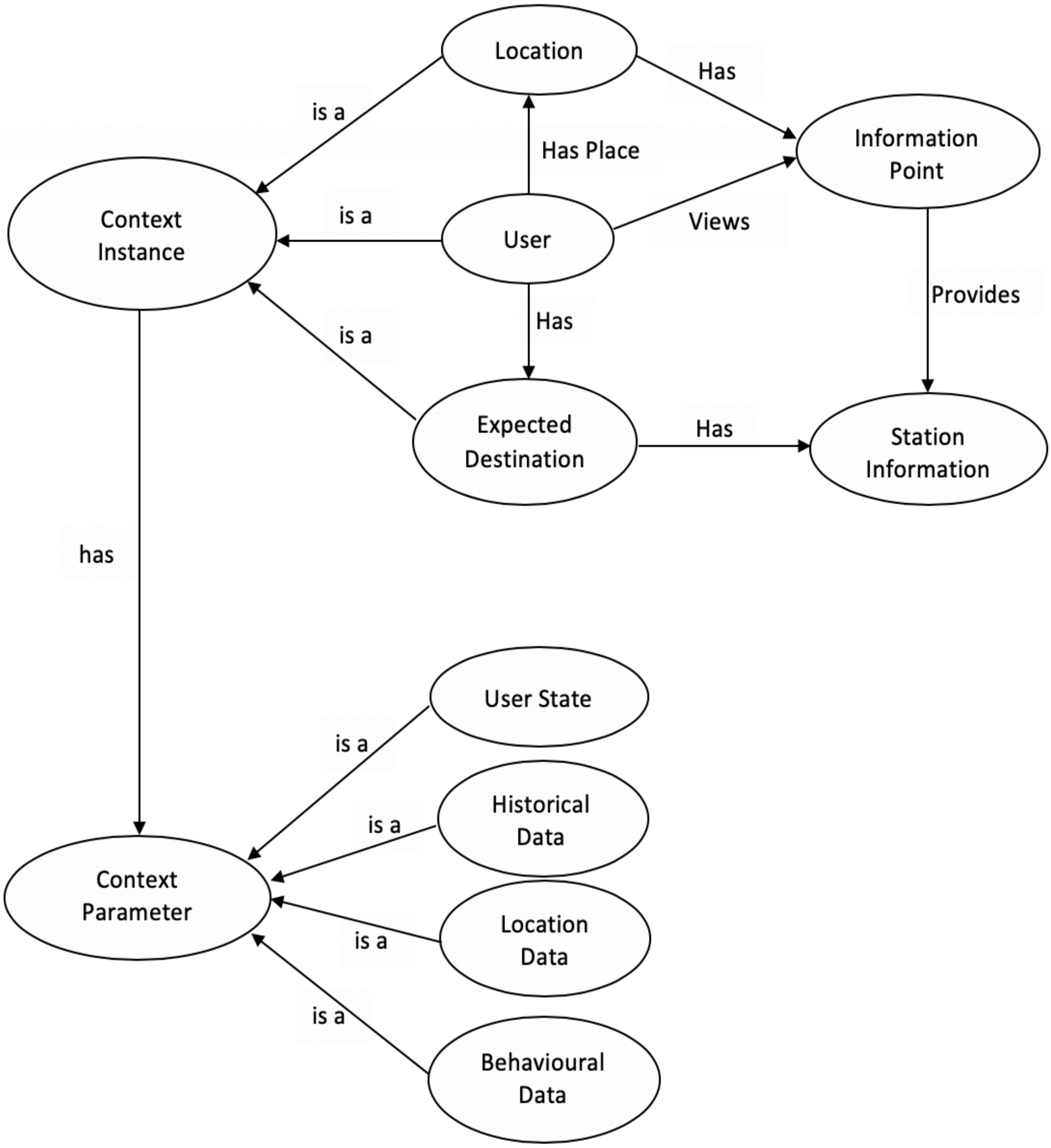


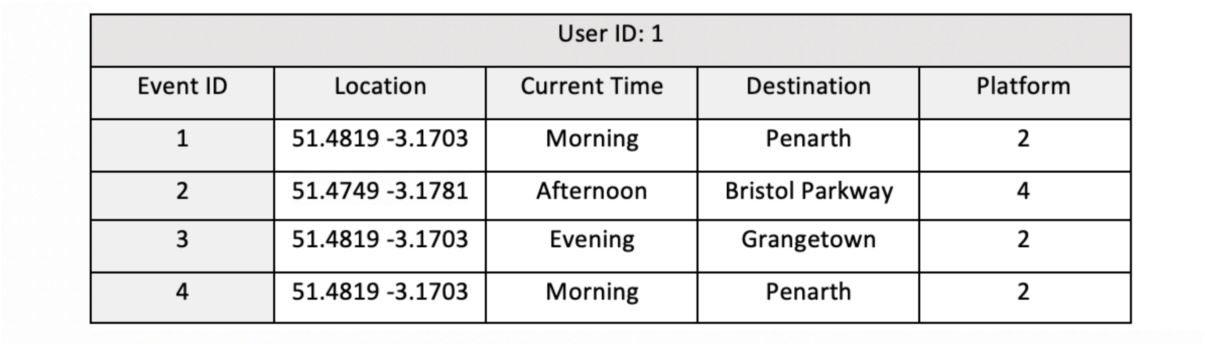
Figure 4: Representing the adopted ontology structure

The surrounding functionality of the application will be used in order to gather the required information, as specified by the ontology. Context parameters will serve to provide assistive information, specifying important data that may be used in order to add further definition to the current environment and place interactions into a greater level of perspective. For example, the analysis of an information point may use historical data concerning previous interactions to determine the most suitable action to take within a given scenario.

More specifically, features of the application such as text recognition alongside information concerning the previous behaviour of the user can be combined in order to identify the required train route information.

The contextual parameters, such as behavioural and historical data, each represent a range of datapoints describing the results of previous interactions. Using this information, a simple probability rules can be applied, assessing the likelihood of a previous destination being reached

The results of such calculation will allow for the generation of parameter values that align with the expected behaviour of the user. In order to store such data, a data structure will be assigned to the user, providing instances of past events in order to derive a probabilistic analysis of conduct. Table 1 represents the table structure that will be implemented alongside the unique parameters.



User ID: 1				
Event ID	Location	Current Time	Destination	Platform
1	51.4819 -3.1703	Morning	Penarth	2
2	51.4749 -3.1781	Afternoon	Bristol Parkway	4
3	51.4819 -3.1703	Evening	Grangetown	2
4	51.4819 -3.1703	Morning	Penarth	2

Table 1: Representing the event table

In order to apply the probability rule, the data will be categorized, providing a basis for computation whilst also maintaining individuality, such that the rule can be applied. Upon the analysis of user location, the current postcode will be gathered when it is determined that the user is at a station. This will serve as the initial filter within the probabilistic system. The current time field, however, will be formatted using a number of key words: Morning, Afternoon and Evening.

Each corresponding to a certain set of hours, these phrases will determine the time of day in which the event occurred. The analysis of the 'Location' and 'Current Time' collectively will serve pivotal in determining the functionality provided. The 'Destination' values serve as key parameters within the event table, providing the corresponding destination arrived at, when provided 'Location' and 'Current Time' as input. If row values are identical, in which 'Location' and 'Time' correspond with multiple rows but with unique 'Destination' values; the most common destination will be employed as the desired destination of the user.

The accuracy of the system will be improved with each new event, altering or affirming the likelihood of a destination being reached. The analysis of information points will serve as input into the event table, determining the destination reached and route taken through the analysis of signs. This approach to learning, although simplistic can serve to achieve comprehensive and reactive results. The addition of rows within the event table will serve to improve upon the data it provides, allowing the system to gain a greater understanding of user preference.

4.6 User Interface Design

In correspondence with the findings gathered in the prior stages of the report. It has been determined that the information provided to the user at a given time should remain minimal and filtered; so not to hinder the view of the user and risk cognitive overload (Katic et al. 2013). As a result, a text-based interface will be devised, serving to highlight important aspects of the environment with small amounts of information placed in close proximity to the information point. Figure 7 represents the design for the text-based interface when an information point (such as a platform sign) has been identified.

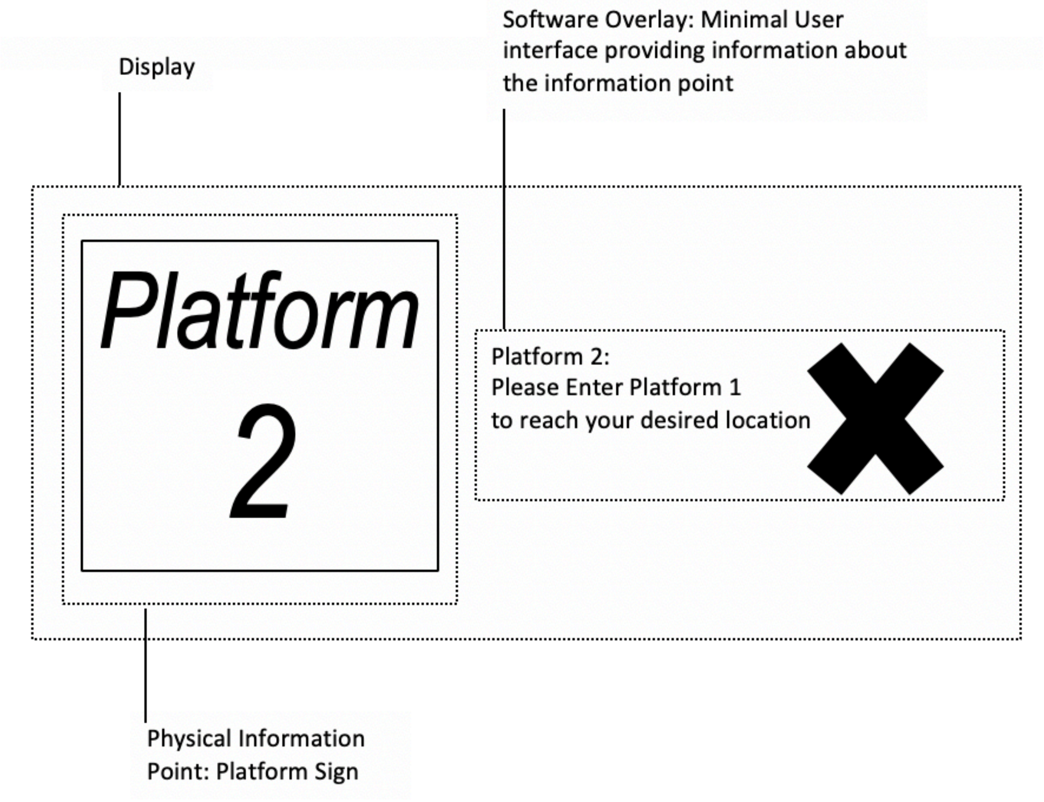

Software Overlay: Minimal User interface providing information about the information point
Display
Platform
2
Platform 2:
Please Enter Platform 1
to reach your desired location
Physical Information Point: Platform Sign

Figure 5: Representing the design of the user interface

As demonstrated in figure 5, the identification of certain aspects of the environment will result in the presentation of a simple interface highlighting only the most important information. As exhibited by (Singh and Singh 2013) the decision has been made to display large visual cues that can be easily recognised at a glance. In the instance described in figure 5, the user is approaching a platform sign that is not applicable in regard to reaching the desired destination. As a result, a large graphic is shown, clearly denoting that another route should be taken.

The creation of information points serves as stimuli, determining the instantiation of further components. This has a range of benefits such as eliminating the need to perform text spotting and thus only performing OCR upon the recognition of certain environmental prompts. Concerning the fulfilment of a broader reaching scenario, the system may need to incorporate text spotting alongside OCR so to determine the most relevant information to the user amongst a wide range of input parameters. For example, a more generally focused contextually aware AR application used by a city commuter will require a wider reaching implementation.

4.7 Supportive Data Gathering

Accompanying information will be gathered through the use of the Trainline website (Trainline, 2019) that are related to the scenario. A range of web scraping techniques can be adopted, serving to parse, locate and return relevant information from HTML code. The site used will allow for the live train and platform information to be provided to the user in the form of a text-based software overlay.

It was initially decided that an API will be used providing a reliable and established connection with the National Rail train time API titled Dawin (National Rail 2015). Despite this, the time constraints (concerning registration and the learning procedure) present in order to adopt this data feed persuaded the adoption of web scraping methodologies.

As a result, web scraping API will be used. The application will employ the rule-based system, in order to determine the desired start and finish locations. Using this information, a search will be performed, gathering the latest departure time and platform data.

5. Implementation

5.1 Environment Rendering

Node objects, as defined by (Apple 2019) within the ARkit SDK, serve to facilitate software components within an AR environment. The frequency and position of nodes within a scene is referred to as a scene graph, determining the coordinates of rendered objects. The scene graph consists of a single root node, outlining the parameters of the scene as a whole.

In order to describe software elements within the scene, child nodes can be implemented with their own coordinate system. The scene graph can be extended in this way so to include a range of nodes, allowing for the facilitation of relational software objects that operate within the same area.

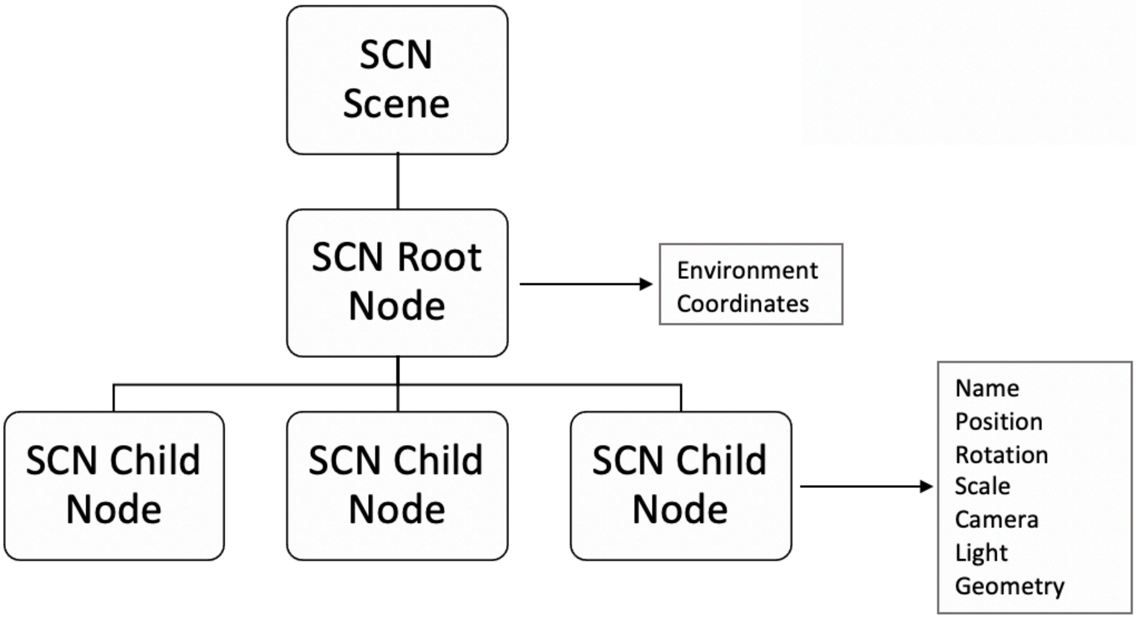


Figure 6: Representing the SCNScene Structure

Through the transformation of such coordinates, the position and appearance of software can be rendered so to fit the application domain and represent a hierarchy of data. Figure 6 represents the relationship between the node objects within an ARKit application and the variable that may be adjusted, detailing the formatting of the node. A node defined using this development method are not representative of a physical object in isolation. Instead variables such as ‘Position’, ‘Scale’ and ‘Light’ serve to describe the appearance of the software component assigned to a given node (Apple 2019).

The SCNNodes implemented, clearly denotes a hierarchical structure and serve as placeholders for information of corresponding importance. For example, the title node is utilised throughout the application in order to represent the most important information, whilst the final node in the SCNScene structure is used to display more variable information to the user such as train times. This philosophy is extended for the remaining nodes. This methodology serves to map structure of the code to the hierarchical positions of the overlay, whist also adhering to the interface designed within the methodology section of the report.

5.2 Environment Recognition

A prominent component of this application is the identification of key elements of the environment. The findings gathered throughout the report informed the development of the method used in order to provide such visual overlays to the user. A set of nodes were devised that adopted a hierarchical structure in order to represent such information.

The recognition of the information points first require the design and creation of reference images. As a result, images of the various information points within a station have been created and implemented within the application. In line with the visual fiducial system specified by (Olson 2011) it is decided that the information points would serve as anchor points for related information, ensuring software overlays are relevant and positioned as expected.

In order to achieve this, the ‘renderer’ function will be adopted. This function (part of the ARKit SDK) allows for relevant information to be assigned and rendered in a fixed state relative to the information point. When appended to platform sign for example, a graphical overlay is presented, providing directional information alongside live train times corresponding to the expected destination (Apple 2019)

It was initially determined that the directional information should overlay the information point. Despite this, and in line with the findings made by (Singh and Singh 2013), it has been decided that the correlation between the software and physical components should be distinct, so not to confuse the user. Figure 7 demonstrates the result of the 'renderer' function upon the analysis of a platform sign.

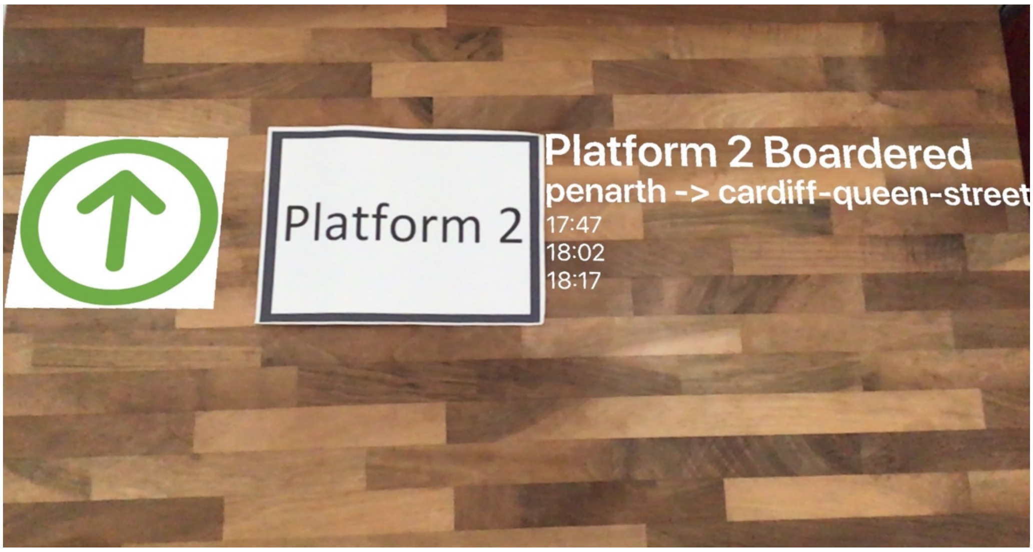


Figure 7: Representing the user interface presented

5.3 Location Analysis

Location Analysis is a functionality adopted by a range of AR applications. In order to gather the current location of the user, Core Location was used (Apple 2019) This API provides a range of functionality allowing for precise control over the collection of information at different durations. It was decided that the location of the user would be gathered at the launch of the application in order to retrieve the postcode corresponding with the station in which the user is located, initialising the generation of an event record. Also, corresponding with the launch of the application, a method is initialised in order to determine the current time and classify it in the form of a string. Corresponding to the time of day (Morning, Afternoon, Evening) the string is matched with the corresponding event value, deriving more context and determining the most likely destination with greater accuracy.

```
func getCurrentTime() -> String {
        let currentDate = Date()// Date method for retrieving time elements from
        calendar
        var calandarComponant = Calendar.current
        let format = DateFormatter()
        format.dateFormat = "HHmm"
        var timeVal: String
        var timeComp: Int
        var timeString: String

        timeString = ""

        timeVal = format.string(from: currentDate)

        timeComp = Int(timeVal)!

        if timeComp < 1200 {
        timeString = "Morning"
        }

        else if timeComp > 1200 && timeComp < 1600 {
        timeString = "Afternoon"
        }

        else if timeComp > 1600 {
        timeString = "Evening"
        }


        print("time:" + timeString)

        return timeString
  }
```

Figure 8: Representing the Get Current Time Method

5.4 Supportive Data Gathering

Determined from the outset to be an important aspect of the application, the gathering and demonstration of relevant information was implemented in order to meet the level of functionality obtained by other AR solutions. The information gathered takes the form of train times relating to the expected destination of the user. In order to gather such information, the Swift Soup package library has been employed, providing a range of functions adopted in order to gather the data provided by a specified website (Swift Soup 2019). The website adopted is that of Trainline.com, chosen as a result of the reliability and corresponding mass adoption (Trainline 2019).

Predetermined parameters were adopted through the use of the location function (retrieving current location) and the probability system, serving to gather the expected destination. Such values that can be utilised to retrieve relevant train time data. A function was devised, taking such variables as parameters, generating a URL specific to the desired route and analysis the appropriate train times. The required URL is produced through the concatenation of the parameters; appended to a pre-determined URL structure. The function used to implement Swift Soup is described in figure #.

In line with findings and the design decisions made within the methodology section of the report, it was determined that the amount of data displayed to the user should be minimised. As a result, the number of route times that are presented to the user during an event has been limited to 3, providing contextually aware information whist remaining unobtrusive within the field of view of the user. Figure 9 demonstrates the positioning of train time information within the user interface.

```
func timeScrape(startPoint: String, endPoint: String) -> [String] { //Returning Arrays of platform
and train times
    var arrivalTime = ""
    let URLString = String("https://www.thetrainline.com/live/departures/" + startPoint + "-to-"
+ endPoint) //Concatinate URL
    let sourceURL = NSURL(string: URLString)
    let html = try! String (contentsOf: sourceURL! as URL, encoding: .utf8)

    do {
      let siteDocument: Document = try SwiftSoup.parseBodyFragment(html)
      let componants = try siteDocument.getAllElements() //Get all elementes ready to be
filtered]

      for i in componants {
        arrivalTime = try i.getElementsByClass("scheduled-time").text()
        break; //Limiting amount of information per fetch
      }

    } catch {
      print("Unable to parse URL")
    }

    let times = arrivalTime.components(separatedBy: " ")
    return times
  }
```

Figure 9: Representing the Time Scape function

5.5 Optical Character Recognition

An important feature of the system, optical character recognition has been adopted in order to determine the possible routes the user may choose to take, thus aiding contextual awareness. The functionality used in order to achieve OCR was incorporated using the Swifty Tesseract package (Swifty Tesseract, 2019)

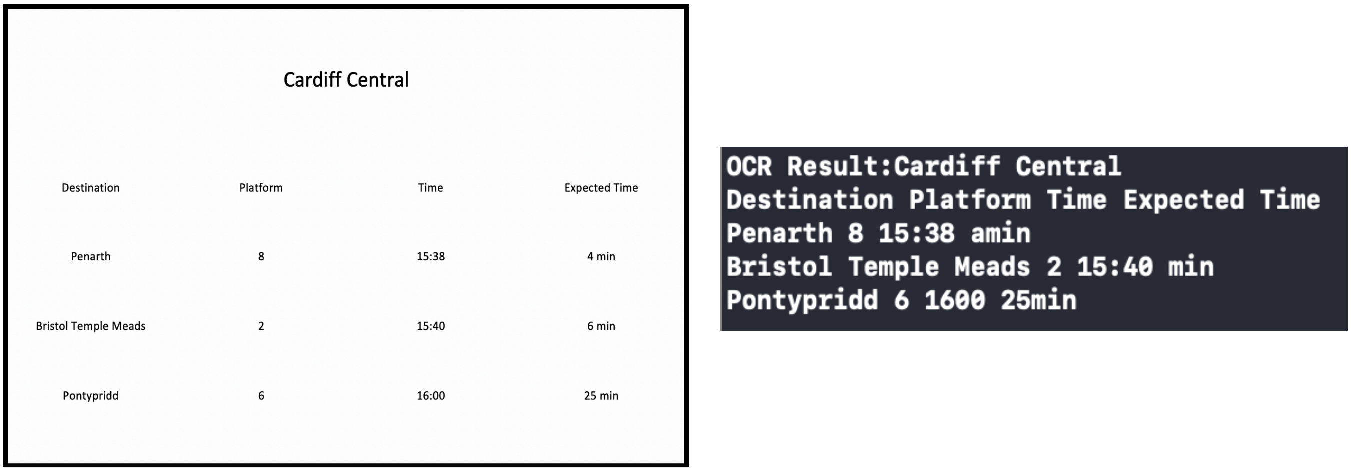
An extension of the Tesseract text recognition engine, the package generates a model that can applied to an image in order to obtain a string of results. Figure # represents the use of the SwiftyTtesseract package and the corresponding OCR function. In this instance, OCR has been used in order to recognise the text within an information point.

```
      let imagefromscene = 
sceneView.session.currentFrame?.capturedImage

            guard let image = UIImage(pixelBuffer: imagefromscene!) 
else { return node } //To be changed to reference image capture
            swiftyTesseract.performOCR(on: image) { ocrResult in 
ocrResultVar = ocrResult!
```

Figure 10: Representing the adoption of Optical Character Recognition

It was initially conceived that the image would recognise an information point and access the image assets within local memory. Despite this, it was important that the application provided real time OCR, so to be able to recognise unique environment properties at different intervals. In order to achieve this, the '.captureFrame()' method, as defined in the ARKit SDK, was adopted. Providing a live image to be processed by the OCR system upon the recognition of a relevant information point (Apple, 2017).



Cardiff Central

Destination	Platform	Time	Expected Time
Penarth	8	15:38	4 min
Bristol Temple Meads	2	15:40	6 min
Pontypridd	6	16:00	25 min

```
OCR Result:Cardiff Central
Destination Platform Time Expected Time
Penarth 8 15:38 amin
Bristol Temple Meads 2 15:40 min
Pontypridd 6 1600 25min
```

Figure 11:The results obtained following the analysis of a train time information point

Upon the unsuccessful recognition of OCR characters, it was determining that, as a result of the nature of the function, the arrays used to store the information may throw an error, relating to the index range. For example, a search for line that has not been identified may return null and will not be stored in the array.

5.6 Contextual Awareness

The development of an intelligent system that is able to determine the likely destination that the user desires is a key component of the application. Through the adoption of a probability rule, the system will be able to provide the best information to the user, based upon a set of previously stored events. It was initially determined that the local database framework shall be adopted in order to implement the event table. Despite this, during implementation of the initial database, it was determined that the use of a local data storage system serves to use a large amount of the device memory.

The implementation of data storage when concerned with a larger application, managing a broader use case, may differ from a local database. For example, many applications of this nature adopt a cloud storage system so to limit local memory uses and query larger sets of information based upon the current scenario. As a result, it was determined that the Firebase cloud database technology would be adopted (Google, 2019).

Employed by a range of consumer facing applications, Firebase allows for integration of a database hosted using cloud technology, provided by Google. Through the implementation of a range of methods, the system is able to utilise the database in order to gather and store events (Google, 2019).

The database is queried at given points within the span of the scenario. Functions are applied to the retrieved information gathered from the location and OCR implementations in order to perform comparisons with the location and expected destination that are detailed within the event table. Using this information, corresponding train time and platform information can be obtained. Figure 12 represents the computation performed by the system.

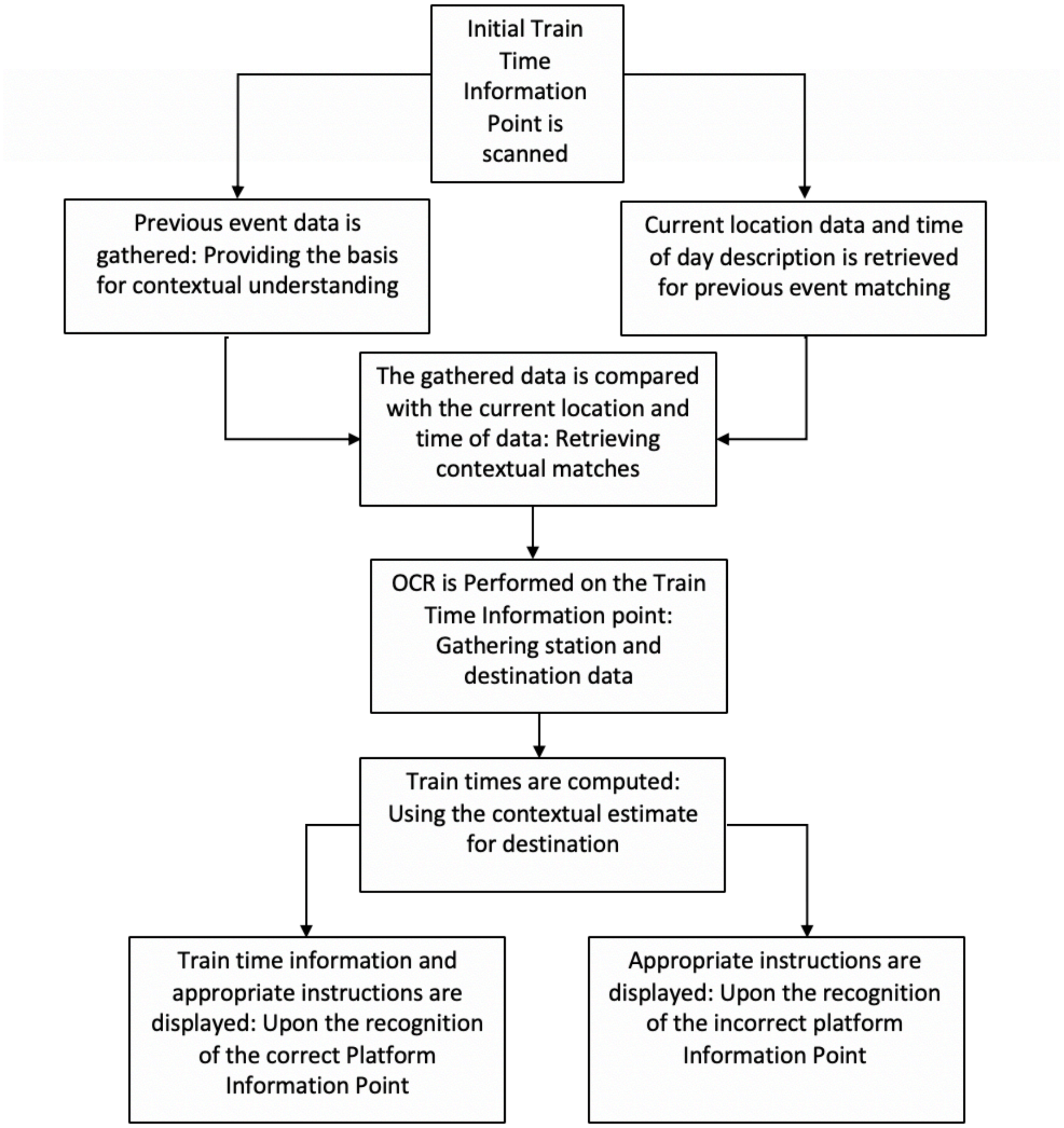

Initial Train Time Information Point is scanned
Previous event data is gathered: Providing the basis for contextual understanding
Current location data and time of day description is retrieved for previous event matching
The gathered data is compared with the current location and time of data: Retrieving contextual matches
OCR is Performed on the Train Time Information point: Gathering station and destination data
Train times are computed: Using the contextual estimate for destination
Train time information and appropriate instructions are displayed: Upon the recognition of the correct Platform Information Point
Appropriate instructions are displayed: Upon the recognition of the incorrect platform Information Point

Figure 12: Detailing the computation performed by the probability system

5.6.1 Consumer Focused Implementation

Deployment at a larger scale, would require a different database structure in comparison to the data storage system employed within this narrow use case. For example, the addition of multiple users would require new database instances so to ensure any suggestions are individual and relevant to the user.

The same concept (deriving contextual awareness from probabilistic calculations) could be applied on a boarder scale with a wider range of use cases. The implementation of such an extension to the system could be achieved through the addition of different database tables, specifying different types of events, such as coffee ordering or walking directions. A further approach could be to anonymise the data, providing a holistic approach that takes into account the behaviours of all platform users in order to determine the most likely scenario.

6. Products

Adopting the findings set out in the Implementation sections of this report, an iOS application was created, utilising the functionality provided by ARkit and a range of APIs in order to meet the criteria set out by the scenario (Apple 2017). A range of methods were devised, each serving to provide unique functionality and complete the required tasks. Such tasks were formulated in response to the completion of background research.

6.1 Requirements

The application was developed in response to the findings made throughout the report. It was determined that the application must demonstrate how contextual awareness methodologies can be adopted in connection with the AR Paradigm in order to improve the utility that is provided to the user. In order to meet the devised scenario a range of functionality has been developed; generating parameters in order to create new events. This serves the development of understanding, predicting user behaviour based upon the results of probabilistic calculation.

This application was developed with the use of a range of APIs, providing the basis for the many different components within this application. OCR has been achieved through the use of the Swifty Tesseract package, a library adopted because of its deep links into the Tesseract Engine (Switfy Tesseract 2019).

Swift Soup has also been utilised, providing a web scraping functionality (Swift Soup 2019). Finally, the data storage system has been implemented through the use of the Firebase cloud storage system provided by Google (Google 2019). Used within the development of many consumer facing applications, it is adopted within this project in order to gather event data in advance of the computation of probability. The application developed is comprised of a range of components, each serving to correlate a unique datapoint.

6.2 Results

In order to determine the effectives of the application a range of steps will be taken. Initially the performance; of the application will be analysed, investigating the effectiveness of each component. For example, an analysis will be performed on ability of the OCR component to read text as desired.

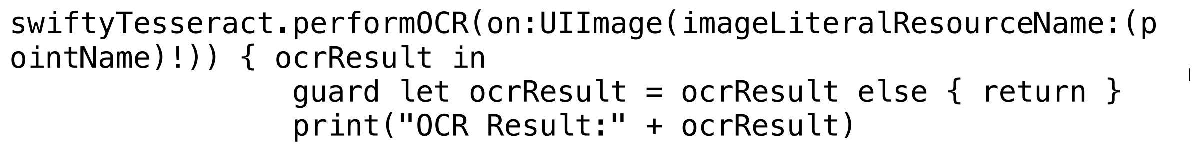
This section will also concern the ability of the program to meet the criteria set out by the devised test case. The results of such analysis will serve to determine the successful creation of a contextually aware AR application whilst also providing a basis for understanding the utility that such a paradigm can provide.

One of the most prominent features of many AR applications is the determination of location. Achieved through the use of Apple's Core Location API, a function was devised that aimed to retrieve the postcode value relating to the current location of the user. In order to test this feature, the function was modified in order to return the coordinates in which the user is situated and gather the required parameter.

It has been determined that this functionality works appropriately, supporting the findings made through the report. It is evident, through the reliable obtainment of such data that the technology often used to achieve location recognition is appropriate for the AR paradigm.

Optical character recognition is a prominent component in the processing of the environment within this application. Through the analysis of information points, text data is gathered in order to determine the possible journeys that may be taken. This functionality has been adopted in order to analyse the text within the environment. However, upon testing the performance of the feature, it has been determined that (if not implemented carefully) any background text within the scene that is not related to the information point may be gathered.

The result of such an implementation may denote a rise in noisy and unfocused data. This was exemplified within the early stages of the project, in which the current frame was captured upon the recognition of an information point. As represented through testing, the reliability of the result is significantly lower than the desired result. The results of such tests led to the implementation of a new approach. In order to overcome this challenge, the stored reference images were analysed, accessing



```
swiftyTesseract.performOCR(on:UIImage(imageLiteralResourceName:(p
ointName)!)) { ocrResult in
                guard let ocrResult = ocrResult else { return }
                print("OCR Result:" + ocrResult)
```

Figure 13: Representing the new approach to OCR

The difficulties of this approach, however, are apparent, implementation of OCR within a broad use case would require a large infrastructure and knowledge of the information points, providing the best conditions for the recognition of characters. The analysis of pre-stored behavioural triggers significantly limits the effectiveness that such contextually aware applications may have.

Environment recognition was adopted within this application in response to the findings made within the research component of the report. In line with the development of the visual fiducial system adopted by (Olson 2011). The implementation of distinct reference images served to inform the application of environmental elements that must be analysed. It was determined that the initial images created (serving to denote different platform signs) were too similar. As a result, duplicate information was provided to the user, limiting the representation of relevant information.

In response to the results of the initial test, the reference images were updated, instead adopting designs of high contrast in order to ensure swift recognition of the information points. Figure 13 represents the results of the initial tests in comparison with the resulting changes made to the reference images. As a result of such changes to the design, the application is able to recognise each information point with a high success rate.

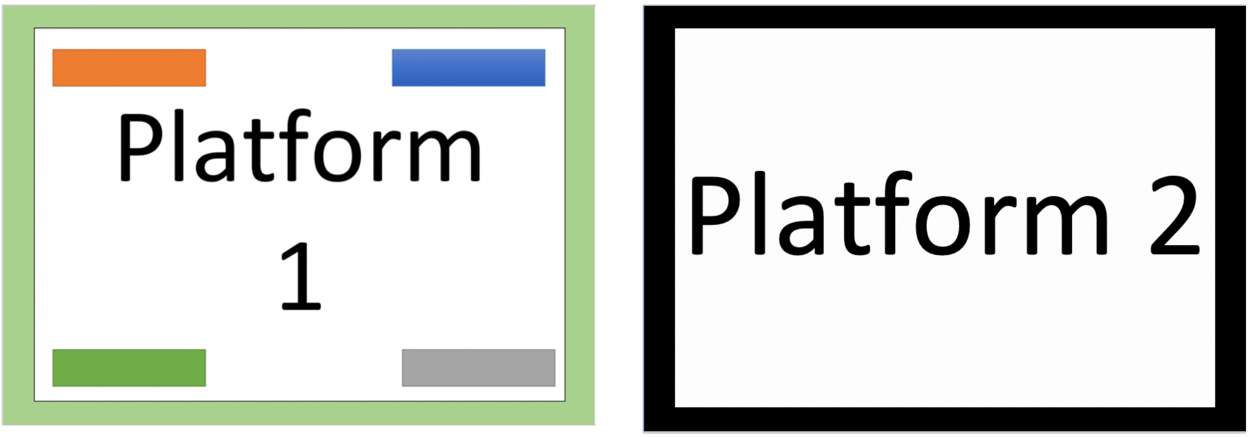


Figure 14: Representing the changes made to the information points

The implementation of an online cloud datastore mechanism in the form of Firebase provided a good method for storing previous 'events' within the created event collection (Google 2019). Despite the utility provided, the asynchronous procedure performed by the data gathering method served unsuitable to the task at hand. The event handler used through the adoption of Firebase served to deprioritise the collection of data until the environment rendering function had been completed. Figure 15 represents the use of the data gathering function in order to gather documents form the event collection field.

The results of the adoption of this function led to the determination that the availability of data, and the flow of application functionality is key in the definition of the context. Information is required in key intervals, in time for the completion of such methods that adopt the data in order to act on pre-determined knowledge. In order to resolve this challenge. the re-implementation of a local database structure in the form of 2D Array was adopted.

```
for document in (snapshot?.documents)! {
    if (document.data()["location"] as! String) == location &&
    (document.data()["time"] as! String) == timeOfDay {
    event.location = (document.data()["location"] as! String)
    event.destination = (document.data()["destination"] as! String)
    event.platform = (document.data()["platform"] as! String)
    event.time = (document.data()["time"] as! String)
    let dest = (document.data()["destination"] as! String)
    destinationArr.append(event.destination)
}
```

Figure 15: Representing the initial method used in order to retrieve data

Environment rendering is the method in which information is displayed to the user throughout the application. The correlation of software overlays with information points aims to provide the user with clear and concise information. This has been achieved. The recognising of information points and reference images initialises the display of visual overlays, as represented in figure 16.

Despite the functionality provided, it has been determined that the unsuccessful recognition of the information points serves to limit the information gathered. For example, in. many cases, the scanning of information points requires close proximity in order to be recognised. As a result, the information provided through such software overlays may not be present when expected, hindering user experience.

Upon the successful recognition of information points, however, the information presented to the user is clear and concise. Appropriate design decisions were made in line with the research findings in order to ensure information, such train times and platform data is managed. This approach ensured that the information remained relevant and non-obtrusive to the user.

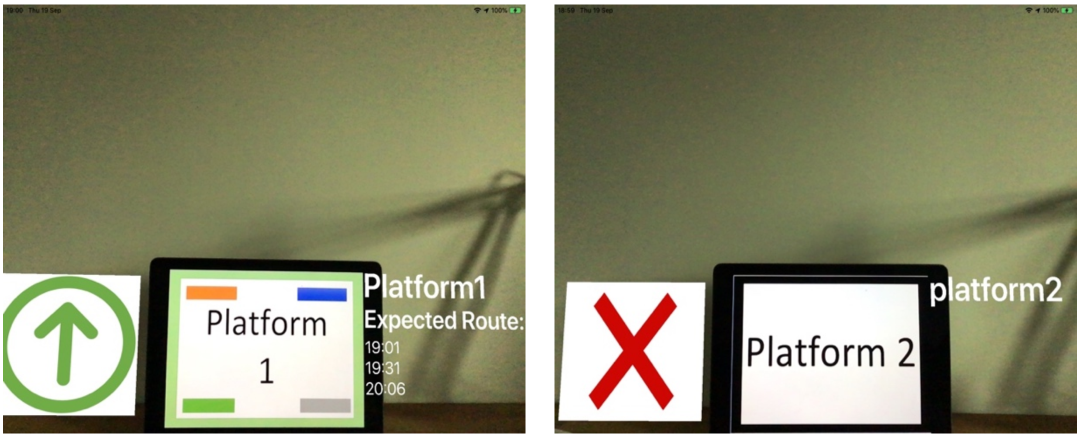


Figure 16: Demonstrating the results of the application functionality.

Despite the effectiveness that such components provide, the accuracy of the data gathered is largely dependent upon the accuracy of previous methods. Looking more closely at the implementation of the final application, for example, it is clear that the determination of the correct platform when analysing the initial information point (the train times signage) enables the computation of the following user-facing instructions denoting platform and route information. Figure 16 represents the recognition of information points and the corresponding functionality. One may wish to analyse the flow chart depicted in figure # in order determine the corresponding computation.

7. Conclusion

This project has explored a range of areas in order to determine the effectiveness of augmented reality applications that facilitate contextual approximation in order to provide relevant information to the user. Various resources were investigated in order to establish an understanding of the components of the AR paradigm whilst also exploring similar projects. The completion of background research provided a foundation to build upon, facilitating the creation of a scenario, outlining a guideline that must be met and a basis for the evaluation of the final project.

The initial stage of this report was to gain an understanding of the Augmented Reality paradigm, investigating the various use cases and the principles adopted in order to meet them. Through the completion of this process, an understanding the typical components and functionality provided by the AR paradigm was developed.

Such analysis achieved at two levels of abstraction, detailing the functionality of the paradigm at a broader level whilst also investigating how contextual awareness has been employed. For example, consumer facing products such as Pokémon Go were explored alongside contextually aware surgical applications used to perform procedures of high risk (Niantic, 2019). Through the completion of this section, the initial aim of the project was met exploring the AR paradigm and its many use cases.

Using the findings as a basis for the development of the application, a range of required components were researched; meeting the second aim outlined; through determining the various methods used in order to implement the functionality that various AR products encompass.

The methodologies used to implement optical character recognition were reconnoitred, for example, in order to determine its uses and the most appropriate method for implementation. Other practices used in order to derive context were also investigated such as the development of a taxonomy used in order to devise a method for determining user behaviour and intention, a concept derived from the findings made by (Tsai and Huang 2018). The analysis of such a component served to aid the fulfilment of a range of objectives and aided the approach taken when developing the application.

Despite the analysis of such procedures used in order to derive context, the greater exploration of other methods could have been performed. For example, an investigation into online learning and other machine learning paradigms such as neural networks may have provided a new avenue to be explored in the quest to explore the derivation of context within AR applications.

The investigation of further implementations, such as location gathering, and environment rendering were paramount to the project. Serving to determine location-based context and the representation of information within the scene respectively. As a result of the research carried out, a range of knowledge was gained, demonstrating a clear understanding of the AR paradigm.

The core aim of the project was to gain an understanding of the effectiveness of contextually aware AR through the development of an application. Using the background research performed as a basis, a scenario was devised; providing a set of guidelines to be met. Determined in order to incorporate the range of functionality often utilised within similar applications, the scenario also provided a basis on which to test the final product, meeting a core objective set out at the inception of the project.

Upon reflection, however, it has been determined that the creation of a broader test case could have been adopted in order to explore a larger use case for AR applications, more clearly investigating various use cases in which the paradigms may serve to benefit the user. Despite this, the scenario developed clearly concerns and allows for the testing of the typical functionality adopted by AR paradigms; whilst also facilitating the analysis of context for the purposes of meeting the determined aims.

The results gathered following the conclusion of development; provide clear evidence concerning the positive effect that contextually aware AR may have. Components such as OCR and the gathering of location, when implemented as elements of the same application are able to derive a range of information key to determining context. This is exemplified throughout the application, with each method utilising previously established data points in correlation with the current situation.

It is recommended that the parameters gathered for a particular use case must be carefully considered; optimising the information provided to the user is dependent on the calculation and filtering of relevant data. In order to determine the information that should gathered, the use case of the developed application should be analysed.

The development of a taxonomy in order to model the data gathered in relation to the behaviour of the user is an important step in the creation of such a system. Future developments in this area should use similar techniques in order to model the adopted approach to contextual awareness.

In order to provide a good standard of utility to the user, it is paramount that the information provided (in the form of software overlays) must remain minimal, so not to overwhelm and provide only the most relevant information. As a result, it is recommended that the design of the user interface is carefully considered, so to remain adaptive and well-suited to the current environment.

It has been determined that the content displayed is more easily regulated in relation to a smaller use case. As a result, the development of such an application used in order to handle a broader user case should incorporate a range of algorithms in order to determine the most useful information to display to the user.

The evident necessity to adapt the reference images for better recognition generates considerable apprehension concerning the ability of contextually aware AR applications to adjust to unique environments. For example, in order to work within multiple scenarios with various information points, the same design methodology must be adopted. The ability of the system to operate within different context and display relevant information to the user may be hindered as a result of such ambiguity. Commercial systems must design around such flaws in order to become viable products for mass adoption.

The development of this project has allowed for a range of information to be gathered, most prominently, it has led to the determination that personalised contextually aware augmented reality technology has a range of potential benefits. This implementation adds significant utility to an existing paradigm.

The presentation of relevant information is well suited to Augmented reality, allowing the user to view data that is in direct correspondence to the surrounding environment. This furthers the understanding of the data displayed whilst also allowing for the computation of different environment objects, through functionality such as OCR. This is exemplified through the recognition of information points. This project has explored the implementation of contextually aware augmented reality technology to good effect, determining how personalised information can be determined and displayed.

7.1 Future Work

An investigation into this topic is multi-faceted and concerns a range of potential areas for analysis. Future analysis into this area should involve exploring new methods for the determination of context. The algorithms adopted within this project in order to derive context are perhaps limited in scope. As a result, other methodologies such as neural networks and online learning should be explored in order to compare the effectiveness of such methods.

Certain constraints that determined the scale of the project, also served to define the scale of the devised test case. Further analysis into this problem domain may also include the development of a broader scenario that must be met by the resulting program. Adjusting the program in order to meet a new specification will serve to determine how the components of the AR paradigm must be adjusted in order to operating within a wider domain whilst still ensuring utility is still provided to the user.

Future investigations into the development of contextually aware augmented reality may also involve the obtainment and analysis of user evaluation of the end product. This will allow for the assessment of different metrics, gaining an understanding of the functionality that may be desired, given unique preferences. For example, a study into the optimum user interface adopted, alongside the number of software components that it is comprised of, can be determined.

Using the scenario in order to determine the approach taken, an application was developed encompassing the various functionality explored throughout the report. It was decided from an early stage that the Swift programming language will be used alongside the ARKit SDK developed by Apple (Apple 2019).

Despite the benefits provided through the adoption of this technology; implementing a tailor-made implementation of tools such as environment rendering; the further exploration of the technologies that are adopted may have provided an insight into different components needed to implement such functionality from the ground up. However, a range of knowledge has been gathered through technology employed. whilst also allowing for the development of comprehension in other areas, such as OCR, through the use of software packages.

8.Reflection

This project has presented a range of learning experiences, regarding both the specific problem domain and the generation of a report that serves to detail the findings made. Perhaps the most prominent learning occurred during the completion of the background research in which a greater understanding of how to complete the research process was gained, obtaining information from a range of sources in order to comprehend contrasting approaches. This is especially prominent, for example when outlining competing products.

A further lesson learned is the importance of the planning. The initial plan submitted to the supervisor of the project was met with feedback determining that the millstones were too broad and instead a further breakdown of the programming components and stages was required. The adjustment of this plan required early preliminary research in order to gain an initial understanding of the programming challenges that would be involved and the technologies that could be adopted to develop the program.

Meeting with the supervisor on a weekly basis throughout the process has served an important device for reflection on the progress made. Providing insight into possible techniques that can be employed during the development of the application, one learned of the importance of obtaining regular feedback at every stage of the project. This aided in ensuring the report remained focused and relevant to the problem domain.

The plan devised at the inception of the project was adjusted significantly throughout. Although still proving an important document, one has learned the importance of creating a plan with contingencies. The development of such a plan in the future will server to diminish the effects of constraints that have occurred as a result of an unexpected learning requirements for example.

Despite the careful planning that was carried out at the start of the project, upon reflection, it has been determined future projects would benefit from the greater implementation of a testing plan, researching different methods for testing the program in a more concrete fashion may have yielded more concrete results for comparison. The development of future projects will adopt this methodology. For example, a questionnaire may have been implemented in order to determine the opinions of the users in response to the effectiveness of contextually aware AR.

A range of lessons have been learned during the completion of this project. Research was carefully structured in order to inform the creation of application in order to meet the devised scenario. The development of a project plan and regular meetings with the supervisor served valuable in keeping the project focused ensuring it is able to meet the aims and objectives.

9. Bibliography

Akbarinasaji, S and Homayounvala, E. (2017) *A novel context-aware augmented reality framework for maintenance systems*. Tehran: Shahid Behesti University

Hilken, T. et al. 2017. *Augmenting the eye of the beholder: exploring the strategic potential of augmented reality to enhance online service experiences. Journal of the Academy of Marketing Science* 45(6), pp. 884-905

Bonnard, Q. et al. 2013. *Designing augmented reality for the classroom. Computers and education* 68, pp 557-569

Schwarz, F. Fastenmeier, W. 2017. *Augmented reality warnings in vehicles: Effects of modality and specificity on effectiveness*. Accident Analysis and Prevention 101, pp. 55-66

Nex, F. et al. 2014. ARCube: *The Augmented Reality Cube for Archaeology*. Archaeometry 57(1), pp. 250-262

Katic, D. et al. 2013. *Context-aware Augmented Reality in laparoscopic surgery.* Computerized Medical Imaging and Graphics. 37(2), pp 174-182

Ong, S.K. et al. 2014. *A context-aware augmented reality assisted maintenance system*. International Journal of Computer Integrated Manufacturing. 28(2), pp. 213-225

IKEA, 2019. *IKEA mobile apps*. Available at: https://www.ikea.com/gb/en/customer-service/mobile-apps/ [Accessed 07 July 2019]

Kipper, G and Rampolla, J. 2013. *Augmented reality: An Emerging Technologies Guide to AR.* Waltham: Elsevier.

Google, 2019. *GLASS: Thank you for exploring with us*. Available at: https://www.google.co.uk/intl/en/glass/start/ [Accessed 19 July 2019]

Huang, W. et al. 2013. *Human Factors in Augmented Reality Environments*. Berlin: Springer

Tsai, C.H and Huang, J-Y. 2018. *Augmented Reality displays based on user behavior.* Computer Standards and Interfaces. 55, pp. 171-181

Kim, K. *et al.* 2016 Understanding users continuance intention toward smartphone augmented reality applications, *Information Development*, 32(2), pp. 161–174. doi: 10.1177/0266666914535119.

Huang, J. et al. 2019. *An augmented reality sign reading assistant for users with reduced vision*. Public Library of Science 14(1).

Niantic. 2016. Pokemon Go. [Application]. Available at: https://www.pokemongo.com/en-gb/ [Accessed: 29 August 2019].

Mehra, P (2012) *Context Aware Computing*. *IEEE Internet Computing*. 16(2), pp. 12-16. doi 10.1109/MIC.2012.31

Grubert, J. et al. 2017 *Towards Pervasive Augmented reality: Context-Awareness in Augmented Reality.* IEEE Transactions On Visualization and Computer Graphics. 23(6), pp. 1706-1723.

Carmaniginiani, J. et al. 2011 *Augmented reality technologies, systems and applications*. Multimedia tools and applications. 51(1), pp. 341-337.

Tsai, C.H and Huang, J-Y. 2018. *Augmented Reality displays based on user behavior.* Computer Standards and Interfaces. 55, pp. 171-181

Singh, M and, Singh, M.P. 2013. *Augmented Reality* Interfaces. Internet Computing. 17(6), pp. 66-70

Lukowicz, P. et al. 2012 *From Context Awareness to Socially Aware Computing.* Multimedia tools and applications. 11(1), pp. 32-41. doi: 10.1109/MPRV.2011.82

Choi, J. et al. 2012 *Organizing and presenting geospatial tags in location-based augmented reality.* Personal and Ubiquitous Computing. 15(6), pp. 641-647

Wang, W. 2015. *Beginning ARKit for iPhone and iPad: Augmented Reality App Development for iOS*. New York, New York: Apress

Olson, E. 2011. *April Tag: A robust and flexible visual fiducial systems.* International Conference on Robotics and Automation, pp. 3400-3407

Siddiqui, J.R. et al. 2013*. Scene Perception by context-aware dominant surfaces.* Seventh International Conference on Signal Processing and Communication Systems.16-18 December 2013. Carrara: Queensland, pp. 1-5

Apple, 2019. *Core Location: Obtain the geographic location of a device*. Available at: https://developer.apple.com/documentation/corelocation [Accessed 18 July 2019]

National Rail. 2015. Darwin. [Application Programming Interface]. Available at: https://www.nationalrail.co.uk/100296.aspx [Accessed: 1 August 2019].
Trainline. 2015. Trainline Home Page. Available at: https://www.thetrainline.com [Accessed: 2 August 2019].

Gubbi, S. and Amruter, B. 2018. *Scene text detection for augmented reality: character bigram approach to reduce false positive rate.* CSI Transactions on ICT. 6(4), pp. 1-322.

Elaayni, I. et al. 2017. *Scene Perception by context-aware dominant surfaces.* International Conference on Wireless Networks and Mobile Communications. November 2017, pp. 1-6.

Behzadan, A. H. et al. 2008. General-purpose modular hardware and software framework for mobile outdoor augmented reality applications in engineering. Advanced Engineering Informatics. 22(1), pp. 90-105.

Kovelov, A. 2007. *On personal-use displays for virtual environments with augmented reality.* Optoelectrics, Instrumentation and Data Processing. 50(6). pp. 549-555

Sassaman, P. 2014. *Context aware augmented reality to assist daily life on a large scale.* IEEE International Symposium on Haptic, Audio and Visual Environments and Games, 11-12 October 2014. Richardson: Texas

Google. 2019. Tesseract. [Source Code]. Available at https://opensource.google.com/projects/tesseract [Accessed: 20 August 2019]

SwiftyTesseract. 2019. Tesseract. [Source Code]. Available at https://github.com/SwiftyTesseract/SwiftyTesseract [Accessed: 20 August 2019]

Kayes, A. et al. 2015. An Otology based approach to Context aware Access Control for Software Services. The Computer Journal. 58(11), pp. 3000-3034.

Google. 2019. Firebase. [Aplication Program Interface]. Available at https://firebase.google.com [Accessed: 1 August 2019]

Chatbi. N. 2019. Swift Soup. [Source Code]. Available at: https://github.com/scinfu/SwiftSoup [Accessed: 30 August 2019]

Sullivan, S. 2017. *Designing for wearables: effective UX for Current and future devices. Sebastopol, California:* O'Reiley.

McCall, R. et al. 2011. Using presence to evaluate an augmented reality location aware game. Personal and Ubiquitous Computing. 15(1), pp. 25-35.

Apple. 2017. ARKIT. [Software Development Kit]. Available at: https://developer.apple.com/augmented-reality/ [Accessed: 11 August 2019].

10. Appendices

1. Notice of Submission form
2. Code Repository for the final application:
   https://gitlab.cs.cf.ac.uk/c1875958/c1875958---dissertation-project