UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE CIÊNCIA DA COMPUTAÇÃO

RENAN LUIGI MARTINS GUARESE

Development and usability analysis of a mixed reality GPS navigator application for the Microsoft HoloLens

Work presented in partial fulfillment
of the requirements for the degree of
Bachelor in Computer Science

Advisor: Prof. Dr. Anderson Maciel

Porto Alegre
December 2018

"Sem GPS pra vitória, cada um faz seu destino."

— DOIDO, CRIOLO

ACKNOWLEDGEMENTS

ABSTRACT

The present work aims to perform a comparative usability analysis between two Human-Computer Interaction systems (HCI) for global geolocation (GPS) navigators. The intent is to compare the conventional use of a navigation application on a mobile device, such as a smartphone attached to the dashboard of a vehicle, to an implementation in Mixed Reality (MR) powered by the Head Mounted Display (HMD) Microsoft HoloLens. By connecting the MR device to a local network routed by an ordinary cellular phone, which is connected to a mobile data network, it is possible to ubiquitously acquire the phone's geolocation data, its magnetometer deviation and a route graph of a navigation Application Programming Interface (API) from its current location to a destination entered by the user. Thus, a series of three-dimensional holograms are created at runtime, geolocated and placed around the user, guiding him through a path indicated on the floor, pertinent to the streets around him that lead to the desired destination. Apart from that, arrows are projected on the way at each crucial point of the path, where some maneuver must be performed, e.g., turning right or taking an exit at a roundabout. In a user experiment, performance and usability were assessed. Results show that users deemed the MR solution to offer a higher visibility both to the oncoming traffic and the suggested route, when compared to the conventional interface, being less attention demanding. EEG readings for most participants also exposed a significantly more demanding focus level for the handheld device. Additionally, an easiness to learn and use was indicated for our system, being almost on par with the already known and highly used application tested.

Keywords: Augmented Reality. Mixed Reality. GPS. Navigator. Microsoft HoloLens. Computer Human Interaction. Usability.

RESUMO

O presente trabalho visa realizar uma análise comparativa de usabilidade entre dois sistemas de interação humano-computador para navegadores de geolocalização global (GPS). Foi almejado comparar o uso convencional do sistema, através de um dispositivo móvel tal qual um smartphone afixado ao painel de um veículo, com uma nova implementação em Realidade Mista potencializada pelo HMD Microsoft HoloLens. Conectando o dispositivo de realidade mista (MR) a uma rede local roteada por um aparelho celular convencional, este conectado a uma rede de dados móvel, foi possível receber ubiquamente os dados de sua geolocalização, de seu magnetômetro e um grafo de rota de uma API de navegação de alta disponibilidade partindo do presente local até um destino inserido pelo usuário. Com isso, é criada em tempo de execução uma série de hologramas tridimensionais geolocalizados ao redor do usuário, guiando-o através de um caminho indicado em seu chão, pertinente às ruas a sua volta que o levarão ao destino desejado. Também são projetadas flechas em seu caminho em cada ponto crucial de seu trajeto, onde deve-se realizar alguma manobra, e.g., dobrar à direita ou tomar uma saída de uma rotatória. Em um experimento realizado com usuários reais, seu desempenho e usabilidade foram aferidos. Resultados mostram que os usuários estimaram que a solução em MR oferecia uma visibilidade maior tanto ao tráfego passante quanto à rota sugerida, em comparação à interface convencional, requerindo menos atenção. Leituras de eletroencefalografia (EEG) na maioria dos participantes indicaram uma demanda significativamente maior de atenção focada no uso do dispositivo móvel. Uma grande facilidade de aprendizado e de uso também foi apontada para nosso sistema, estando quase a par da aplicação móvel altamente conhecida e usada.

Palavras-chave: Realidade Aumentada. Realidade Mista. GPS. Navegador. Microsoft HoloLens. Interação Humano Computador. Usabilidade.

LIST OF FIGURES

LIST OF ABBREVIATIONS AND ACRONYMS

|	
2D	Two-Dimensional
3D	Three-Dimensional
API	Application Program Interface
AR	Augmented Reality
BCI	Brain-Computer Interface
EEG	Electroencephalography
GPS	Global Positioning System
HCI	Human-Computer Interaction
HDD	Head Down Display
HL	Microsoft HoloLens
HMD	Head Mounted Display
HTTP	Hypertext Transfer Protocol
HUD	Head-Up Display
MR	Mixed Reality
PND	Personal Navigation Device
SD	Standard Deviation
SUS	System Usability Scale
TCP	Transmission Control Protocol
UI	User Interface
TLX	Task Load Index
UWP	Universal Windows Platform
UX	User Experience

CONTENTS

1 INTRODUCTION

In 2004, the mobile phone industry began to incorporate GPS receivers into their devices to support law enforcement with locations for emergencies (WHIPPLE; ARENSMAN; BOLER, 2009). Since then, smartphones equipped with a navigation system got disseminated throughout the world and are widely used to guide drivers to reach their destinations. When using a Personal Navigation Device (PND) while driving to follow a suggested route, vehicle operators have to often take their eyes off the road ahead onto the GPS navigation screen. These actions may lead to accidents, severe injuries and even death. Studies have shown that distraction by a navigation device was significantly associated with the most serious incidents, pointing out that among the 44 deadly incidents researched, 21 involved distraction (LIN et al., 2017). In five of these cases, people were using non-critical features of their navigation device. Likewise, amid the usual causes of the accidents investigated, the authors indicated that over 24% of the total incidents regarded missing road characteristics, while an issue with routing guidance, either in visual or audio form was present in 16% of the incidents.

Besides exposing that drivers spend less time looking at the road ahead when they have a traditional PND, another previous work (KUN et al., 2009) also stated that glancing at the visual display did not prove to be necessary to complete a navigation task, since no subjects in their tests missed directions for any of the navigation aids tested. In an investigation comparing three different interface configurations of a PND, namely purely visual, audio-visual and purely audio (JENSEN; SKOV; THIRURAVICHANDRAN, 2010), the authors recorded several incidents (e.g. running red lights, missing turns or speeding violations) while participants glanced at the GPS system. The authors also stated that audio only participants performed better in relation to their driving performance than the other subjects. Regarding interface issues, another work (BROWN; LAURIER, 2012) identified that excessive instructions are a significant problem with GPS usability. In another study (KUN et al., 2009), even though GPS navigation systems with screens have proven to be too distracting while driving, participants still answered that they prefer to have a GPS screen for navigation. The authors hypothesized that users may seek reassurance that they are on the right path, citing that they might get anxious that, for example, they missed a turn and received no feedback.

All of the aforementioned works inspired the intent of supplying this continuous reassurance and also reduce the need to take the driver's focus away from the road. Re-

garding this, the present work presents the development of a GPS navigation system using virtual elements placed at the surface of the road using Mixed Reality (MR). The more comprehensive term MR is used in this work instead of Augmented Reality (AR), which is more generally applied. The MR portion of the Reality-Virtuality continuum (MIL-GRAM; COLQUHOUN, 1999) covers essentially the entire breadth of the spectrum, but it also excludes the end points. Thus, any application that mixes the real environment with at least some virtual components, but never occluding the full vision of the user, can be considered MR.

Figure 1.1: Definition of Mixed Reality, within the Context of the RV Continuum.
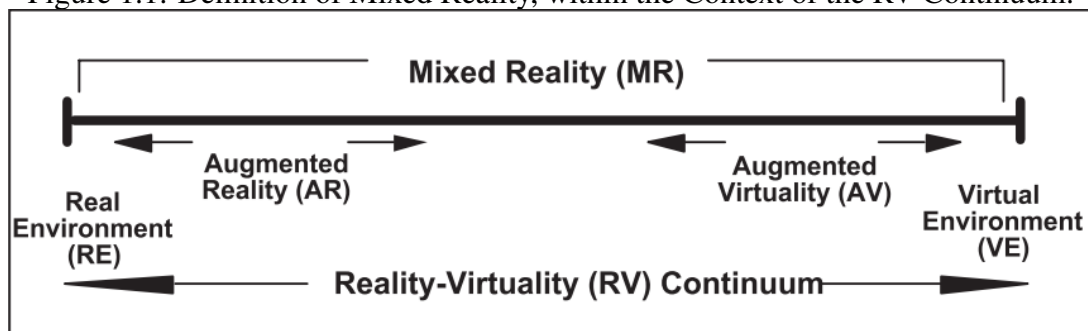
Source: (MILGRAM; COLQUHOUN, 1999)

With the help of these virtual elements through MR, users can be guided throughout the extent of a route without the need to take their focus away from the road. A similar goal is fulfilled with Head-Up Display (HUD) navigators, where important parts of a route are projected onto the windshield of a car (MAROTO et al., 2018). Such interfaces have proven to produce better speed control and faster reaction to both speed limit signs and urgent events than regular Head Down Display (HDD) interfaces (LIU; WEN, 2004). The main deviation from these HCI systems to the one proposed here is the presence continuity provided by the context-aware elements in MR. By projecting a path onto the surface of the road, the MR objects can be placed on top of their proper geo-located analogous places, such as minimal curves of the road, round-abouts and corners. Such details are not present in HUD navigators; they only expose the main maneuvers a driver should perform. HUD systems also require a windshield to be projected on, which is enough to exclude motorcycle and bicycle riders from their use. This also affects the usage between daytime and nighttime use, since the projection may be affected by sunlight glare (HUDWAY, 2018).

2 RELATED WORK

In the present chapter, a series of studies and products that are somewhat similar to or even inspired the present work will be described. Their similarities and divergences will be mentioned, as well as their qualities, design choices and observed flaws.

2.1 Bycicle and motorcycle navigation systems

Dancu et al. (DANCU; FRANJCIC; FJELD, 2014) studied two alternative solutions for navigation using a map while cycling in an urban environment: firstly a standard smartphone display attached to a bicycle handlebar and, secondly, a GPS-based map navigation using a bike-mounted projector named Smart Flashlight. The latter projects on the road a map with the requested route, as shown in Fig. 2.1, much like in a regular PND. A few points comparing the usability of the tested solutions were inquired of the six subjects that complied to their tests. Most of the subjects preferred the projection system in terms of easiness to use, safety, helpfulness, fun, traffic visibility and route attention. Regarding their original solution, when compared to the standard one:

- 69% of subjects claimed it was easier to use;
- 75% found it safer;
- 69% stated it helped them more regarding support for navigation;
- 100% claimed it was more fun;
- 63% of subjects perceived route attentiveness as being higher;
- 69% reported road and traffic visibility as being higher.

Apart from only displaying a map, the Smart Flashlight is highly affected by external light sources, such as street lights and car headlights. Overtly, it only works at night.

In another work regarding unusual navigation techniques, the authors tested two methods using spatialized music to guide pedestrians and cyclists toward a destination without any prior training (ALBRECHT; VääNäNEN; LOKKI, 2016). Participants had to follow the apparent three dimensional origin of the sound they heard in headphones, taking advantage of the natural ability of humans to localize the source of a sound. The authors tested two conditions: route guidance and beacon guidance. The first method leads the user along a route with the audio source a certain distance ahead. The second tested

Figure 2.1: Bicycle equipped with the Smart Flashlight.

Source: (DANCU; FRANJCIC; FJELD, 2014)

approach places the music origin in each step of the way, using turn-by-turn guidance. Route guidance represents a more seamless experience, where it constantly moves along the route some distance ahead of the user instead of "jumping" from waypoint to waypoint. Both techniques were deemed good alternatives by the subjects. Beacon guidance was generally considered to be suitable for familiar surroundings, while route guidance was seen as a better alternative for areas that are unfamiliar or more difficult to navigate. Regarding the main flaws of these methods, the authors mention that using headphones as a pedestrian or cyclist may result in attentional blindness and environmental isolation. The former is due to the cognitive distraction of interpreting auditory stimuli and possibly manipulating electronic devices, reducing mental resource allocation or attention to outside stimuli. The latter represents the inability to hear sounds from the surroundings due to attenuation caused by the headphones and masking caused by the sounds reproduced with the headphones.

In regard to innovative GPS systems for motorcyclists, Kiss et al. (KISS et al., 2018) introduced MOVING (MOtorbike VIbrational Navigation Guidance), a kidney belt that conveys navigation cues through 12 vibration motors. Their evaluation with 16 participants exposed that their system outperforms commercial navigation systems in terms of turn errors and distraction. Apart from these positive results, 15 of the 16 participants

stated that it was easy to differentiate different vibration patterns from each other while riding the motorbike. All subjects also claimed that the visual systems are more distracting than the haptic system.

2.2 Indoor Navigation Systems

Either by using solely a smartphone, its accelerometer and gyroscope data, the Google Glass HMD or the Microsoft HoloLens, several indoor navigation studies have been accomplished. The most recent ones present AR-based indoor navigation applications that use pre-scanned environmental features and markerless tracking technology to guide users inside buildings, such as (DEY; KARAHALIOS; FU, 2018), (REHMAN, 2016) and (BåGLING, 2017). Although interesting and valid tests, the spatial mapping of all of the aforementioned works had to be performed *ad hoc*, since a generalized mapping system for the insides of buildings does not exist. Rehman (REHMAN, 2016) claims that the 3D scanning during the pre-deployment stages was time consuming and complicated, which hampered their ability to conduct large scale tests. Aside from that, the same study exposes that, when compared to non-AR solutions, subjects using the AR solutions presented the lowest workload in their tests. Regarding the HoloLens (HL) device specifically, Bågling (BåGLING, 2017) mentions that their approach showed that people in an indoor finding objects task performed better using the HL than using a smartphone and paper maps. Test subjects also pointed out that it was a technology they would like to see more of in the future.

2.3 HUD Systems

As was previously mentioned, HUD navigation systems already prevent car drivers from glancing away from the road to get route information. Such systems have increased in popularity in the last few years (MAROTO et al., 2018). Either as built-in systems in specific car models or standalone solutions that could be incorporated in most cars using a smartphone (SYGIC, 2013; HUDWAY, 2015; GARMIN, 2013), they are not MR systems. In addition to the aforementioned sunlight glare issue (HUDWAY, 2018), these solutions do not present a fully immersive setting, since their projections are static, two-dimensional and do not follow the details of the road ahead, such as curves, ascents and

Figure 2.2: Sygic HUD.

Source: (MAROTO et al., 2018)

descents.

Figure 2.3: AR PND from the driver's perspective.

Source: (MEDENICA et al., 2011)

Prior to these products, (MEDENICA et al., 2011) compared the use of an AR HUD PND with two HDD PND methods by using a high-fidelity driving simulator: the first a street view PND, i.e., a device that presents, apart from the route, a visualization of the street from the driver's perspective; and secondly a standard PND. A thorough usability study was made. The results exposed that the HUD option provided for more visual attention at the road ahead as compared to both HDD devices. On average, when using an AR PND, participants spent about 5.7 sec and 4.2 sec more each minute looking at the road ahead in comparison to the street view and standard PNDs, respectively. Their

AR solution, besides being purely simulated, presented a yellow path projected above the center of the road at a height of about 2 meters, as depicted in Fig. 2.3. In our work, we reproduced their simulated AR solution in a real world MR setting.

       This chapter regarded some products, studies and projects that presented a similar theme to the work at hand. Most of these exposed relevant details for the development of this project. The subsequent chapter aims to debate over such development, mentioning the tools used, the design and implementation processes, the final prototype and all tests performed in order to legitimize its usability.

3 METHODOLOGY

In the intent of disclosing the methods used in this project, the current section will present the conception, development and assessment procedures conducted. The research done, development tools used, implementation methods taken and test protocols applied are described and discussed.

3.1 Development tools

Concerning the stage prior to development, a set of tools were researched and chosen as the most appropriate for the project. These were selected for being judged by the authors as either the most advanced in their area or the most highly available to users.

3.1.1 Microsoft HoloLens

In favor of offering an Augmented Reality experience, users are required to use some sort of spatially aware display. Arguably, the most interesting AR experiences involve markerless tracking, i.e., tracking the environment around the user without the need to place obvious markers in their current field of view or rely on an external tracking system. Moreover, a GPS navigation feature is meant to work ubiquitously, making it intrinsic not to require markers. All of this can be accomplished with the Microsoft HoloLens (HL)[1]. Launched in 2016, it is to this date the most advanced commercially available mixed reality HMD. It is capable of tracking the space around the users, creating a 3D mesh of the local environment and using it to occlude virtual objects with real ones elegantly. As its main interaction strategy, the HL displays a virtual cursor placed at the center of the user's field of view and recognizes certain hand gestures such as the *tap*, used to "click" on buttons, and the *bloom*, usually performed to open a menu or close applications. For the combination of its main features, the HL was chosen as the MR platform for the ongoing work.

---

[1]https://docs.microsoft.com/en-us/windows/mixed-reality/hololens-hardware-details

3.1.2 Unity 3D Engine

Commonly used in the commercially competent gaming industry, the Unity 3D engine[2] offers a plethora of visual and simulational assets regarding 2D and 3D development. It would suffice to say that it is also recommended by Microsoft for developing HL applications[3], even though it is technically possible to work solely with Microsoft's own Visual Studio[4]. In addition, prior experience with Unity, its ease of use in three dimensional environments, as well as the possibility to export to different systems, were the definitive reasons for choosing it as the main development platform.

3.1.3 Google Maps Platform

For a GPS navigation application to be complete, it requires both geo-located road mapping data and routing algorithms. The Google Maps Platform[5] provides that with not only simple to use information, but also in different formats and languages. It is also the most openly available and complete mapping API, additionally supplying information such as traffic, altitude, local businesses and satellite images. In the most simplistic form, by inputting an origin and a destination, the API returns to the user a list of geo-located coordinates forming the suggested route, providing additional information in each node, such as lists of nodes designing the geometry of all segments of the road (polylines) and instructions for each turn, amounting to a graph of data. With this in mind, and for its already well-established tradition in the field, the Google Maps Platform was chosen as the main database for this project.

3.1.4 Android Operating System

Since the HMD chosen does not provide all hardware necessary for the application at hand, a second device had to be used. The components that were still missing for the task are a GPS receiver, a magnetometer sensor and any source of internet connection that could be accessible from city streets, such as cellular data, for instance. All these

---

[2]https://unity3d.com/
[3]https://docs.microsoft.com/en-us/windows/mixed-reality/install-the-tools
[4]https://visualstudio.microsoft.com/
[5]https://cloud.google.com/maps-platform/

elements are commonly present in any conventional smartphone. Thus, two main options seem discernible: Android OS[6] and iOS[7] phones. Unity can export applications for both of them. Given the convenience for developers and users alike, the Android OS was chosen as the default option for our tests since its applications are objectively simpler to deploy. After all, iOS developers are required to use a macOS[8] computer, as well as being part of their developer program[9].

3.2 Design and Implementation

In this section, the concept, design and implementation processes of the actual application the present work has proposed will be discussed.

As previously mentioned, the original motivation came from the frequent need to take the focus of a car driver or a motorcycle rider away from the road and onto a GPS navigation device, which is usually placed either in the car dashboard or in the motorcycle handlebar. After considering all the points mentioned in the introduction, the main goal was to design and develop a mixed reality solution that could guide users through a route using GPS tracking and navigation services. Since a regular smartphone was used to transmit GPS data to the HL, this transmission channel was also used to provide the phone's magnetometer deviation and access to the Google Maps Platform. In this manner, two applications were developed[10]: firstly, an Android app that acts as a server, broadcasting information; secondly, the HL application itself, presenting navigational holograms to the user, and indicating the path to be followed. Both programs will be discussed in separate subsections.

3.2.1 holoServer: an Android application

Since the Microsoft HoloLens lacks a GPS receiver, an Android phone was chosen as a means to transmit geolocation data. Even though the HoloLens does have a magnetometer, its data is not available to developers as of this date (HOLOLENS, 2016). Thus, the smartphone compass data was used in order to properly rotate the route to match with

---

[6]https://www.android.com/
[7]https://www.apple.com/ios/ios-12/
[8]https://www.apple.com/macos/what-is/
[9]https://developer.apple.com/programs/
[10]https://bitbucket.org/renghp/holonav/src/fixedHoloApp/

the real world analogous streets, i.e., aligning the hologram's north to Earth's north. Regardless of the application itself, the cellphone also routes its local network to the HL, since an ubiquitous internet connection is required to acquire routing information from the Google Maps Platform while on the street and the HL is not mobile data network (e.g. 3G and 4G) compatible.

Figure 3.1: The holoServer app.
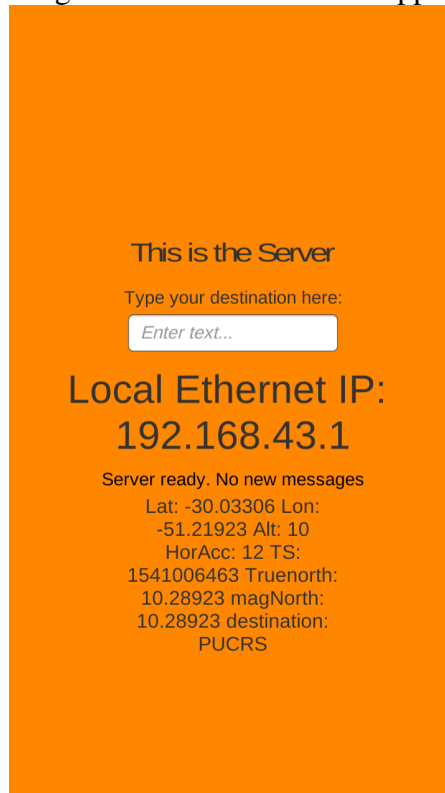
Source: Authors

As can be seen in Figure 3.1, holoServer is a very simple app design-wise. Apart from a text box meant for the user to input a destination, it is solely composed of textual information. It displays:

- the phone's local IP - for debugging connection purposes;
- a connection status message;
- the phone's current GPS location, e.g., latitude, longitude, altitude, horizontal accuracy and a time stamp, supplied by its GPS receiver;
- the phone's deviation, in degrees, from Earth's magnetic north, supplied by its magnetometer;
- the user inserted destination.

The android application is merely a way to supply the HoloLens with features it

does not offer. Ideally, a single MR headset device should suffice for our project design. Nevertheless, a single design choice was made against this principle: typing the destination. Typing with the default on-screen keyboard in the HL is not a pleasant experience (MICROSOFT, 2018). To simplify the input method, as well as allowing a better user experience (UX) for the current version of the application, a choice was made to let users type their destination input on the phone, using a regular smartphone keyboard.

Regarding the development process *per se*, a TCP server is created by the app using the *server* script, along with a TCP Listener and a list of clients. As default, any client in the local network is accepted into the client list via port 6321. Once a client is connected, the server will start broadcasting a message containing the aforementioned textual information displayed on-screen. All of these pieces of information are updated by the *gpsSignal* script, which starts the location and compass services and updates them with a requested accuracy of 1 meter. This accuracy could be less precise when targeting fast moving vehicles. Our primary tests, however, had to be done on foot, so a higher accuracy was necessary.

3.2.2 holoNav: a Microsoft HoloLens application

Having solved the missing components problem, the HoloLens suffices as the hardware necessary to build an immersive MR navigation application. Developed by the authors, the holoNav is the proposed solution to this work's focus problem. It was developed using the Unity 3D Engine with C# scripts and exported onto the HL HMD using the Microsoft Visual Studio. This MR application communicates with the previously exposed holoServer in order to download a route graph that leads users from their current location to a requested destination.

Based on the mentioned graph, the holoNav generates virtual holograms such as a series of two-dimensional rectangles - serving as a ground-level path - and three-dimensional arrow objects, guiding the user along the path in a turn-by-turn manner. It maps the latitude and longitude coordinates of these objects onto the 3D space around the user, placing the generated route in a rather precise manner, according to the real analogous 2D distances between these points in space, as depicted in Fig. 3.2. In order to finish the proper mapping, the use of the magnetometer deviation readings from the holoServer app is required. Given a proper positioning of the device during a short calibration process, it is possible to rotate the generated holograms using the initial user position as an

axis, adjusting their rotation to match the real positions of objects around the subject. This process aligns the simulated "north" direction of the holograms with the magnetic north of the Earth.

Figure 3.2: Examples of routes generated by the holoNav application.

Source: Authors

3.2.2.1 Connecting to the server

In regard to the proper development process, after having connected the HMD to the Android phone's local network, the HL application uses the *clientDualSystem* script to connect to the proper TCP port using sockets. The IP and port used are preset to match

the holoServer's. However, they can be modified in runtime with a virtual holographic numeric keyboard by the user to connect to another source. As reported by (KUZHA-GALIYEV, 2017), implementing a TCP client that works in both the Unity editor (for development) and UWP on HoloLens (for production) is troublesome, since both only accept certain terminologies for the same processes and regard the other's as syntax errors. Thus, it is necessary to continuously use *if* statements to check which of the two platforms the code is currently being parsed on or run in, hence the "dual" in the script's name. By doing so, all of the previously indicated information displayed in the holoServer app is received in the HL application via broadcast, updating continuously.

3.2.2.2 Route graph

Requesting a route graph to the Google Maps Platform Web Service is simple, since plenty of information and methods are available (GOOGLE, 2018). After creating an account and using its API key, one may request a route using an HTTP request by providing it with origin and destination points, in either coordinates or proper words (such as the name of an University or its address) and an optional navigation mode, e.g. driving, walking or cycling. Other parameters are accepted. This work, however, focuses on keeping the navigation simple as to avoid a complex UX, so these will suffice. By requesting so, a response containing vast information regarding the suggested route is returned, either in JSON or XML formats. Both formats exhibit similar content, only their syntaxes differ, so neither will be discussed in depth. In order not to pertain to either format, regular expressions were used to parse their contents into usable data in the *googleAPI* script.

From the response file, the folowing pieces of data are extracted in order to create and manage the virtual route:

- the initial geographic position, in coordinates;
- the destination geographic position, in coordinates;
- the full length of the suggested route, in kilometers;
- an estimated duration of the route, in minutes or hours;
- an *overview polyline*, summarizing the entire length of the route in an encoded string;
- every node of the route, each representing a step of the journey, i.e., a *turn* the user has to take in order to follow the path. Regarding each node, they contain:
    - its coordinates;

- a set of instructions, describing what the user should do in the current step of the graph;

- a *maneuver*, summarizing the instruction in a keyword;

It is important to mention that the *maneuver* field is only present in some nodes. Most of these attributes will be discussed in depth in the following subsections.

3.2.2.3 Mapping coordinates to distances

Several methods exist to calculate great-circle distances, i.e., distances between two pairs of latitude and longitude coordinates, such as the Haversine Formula (INMAN, 1835). These come attached to their particular computational cost and its usually related accuracy. In the present case, this operation needs to be performed very often and with a highly volatile number of points in a graph, since the user requested route can vary tremendously in length. For this reason, this method was simplified in order to reduce the overhead of the algorithm based on the notion that one degree of latitude can be roughly approximated to 111111 meters (BEDING, 2016). The impact of this simplification on accuracy will be discussed in a posterior section. In this manner, the two functions present in the algorithm below were used to calculate distances regarding their latitude and longitude coordinates.

Algorithm 1 Calculating the approximate vertical (Y-axis) and horizontal (X-axis) distances between two pairs of global coordinates, respectively. Based on (BEDING, 2016) and (INMAN, 1835) approximations.

1: **function** DEGREESLATITUDETOMETERS($latitude, originLatitude$)
2:     $displacementLat \leftarrow latitude - originLatitude$
3:     $distanceY \leftarrow displacementLat \times 111111$
4:     **return** $distanceY$
5:
6: **function** DEGREESLONGITUDETOMETERS($longitude, originLongitude, latitude$)
7:     $displacementLon \leftarrow longitude - originLongitude$
8:     $distanceX \leftarrow displacementLon \times 111111 \times \cos(\pi \times \dfrac{latitude}{180})$
9:     **return** $distanceX$

Using these functions, it is possible to use the current GPS position of the user as an origin point and map all points of a received route graph onto the X-axis and Y-axis distances away from the user, in a two-dimensional manner. The lack of precision in this method accumulates over distances, however. The farther the points, the larger the error. In the interest of compensating for this, as the user moves and the current GPS
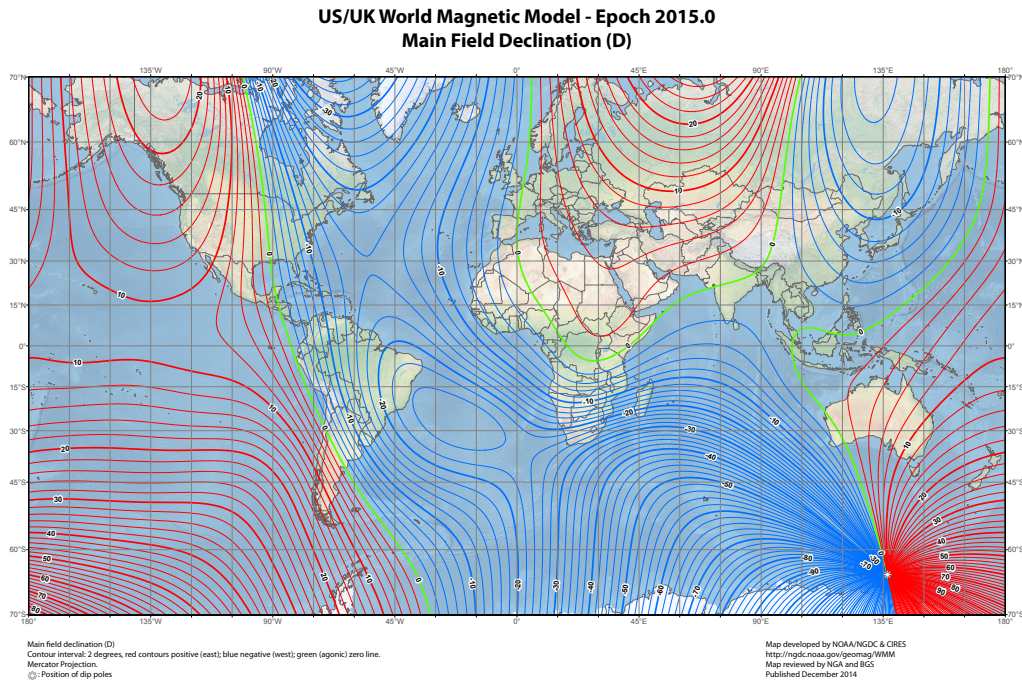
coordinates change, these distances are all recalculated, keeping the error minimal for all of the closest, and visible, objects.

3.2.2.4 Rotating holograms according to Earth's north

Once mapped onto a 2D space, even though the distances between every object are virtually correct, the holograms are facing an arbitrary direction, based on the direction the user was facing when the route was requested. In order to position them accordingly to their real analogous points in space, e.g. placing the virtual route on top of the actual road, a rotational transformation is required. During the positioning of the path holograms, the exact front of the user was assumed to be the north direction to facilitate the mapping. In this fashion, the necessary rotation is equivalent to an angle α, which represents the number of degrees this direction is divergent from the factual north. By receiving the magnetic north heading in degrees available from the smartphone magnetometer, a good starting point measurement is already available to find α. Such device, however, needs to be pointed exactly at the same direction the user was facing previously. On top of that, the magnetic declination, i.e. the deviation between the true north pole of the Earth and its magnetic equivalent is not constant across the Earth (CANADA, 2018). This deviation not only follows quite a peculiar pattern globally, as can be seen in Fig. 3.3, but it also changes along time (NGA; DGC, 2015).

Since this deviation angle does not change drastically in less than a couple of hundred kilometers and in order to proceed with the development and tests of the present work, a transient design choice was made to keep this value *ad hoc*, using the angle of $16^o59'$, which pertains to the city of Porto Alegre, Brazil. As to facilitate with the particular position the user has to hold the phone for the magnetometer to present an optimal reading, a hologram in the form of a red arrow pointing ahead is placed in the scene, as showed in Fig. 3.4. This procedure is managed by the *magnetometerController* script. The user is suggested to position the phone according to this arrow. This process has to be done once every time the application is run, during the calibration phase, since the HoloLens' embedded accelerometers and gyroscope ensure that all object positions and rotations will be kept untouched unless otherwise commanded via code.

Figure 3.3: Map of magnetic declination in 2015.

US/UK World Magnetic Model - Epoch 2015.0
Main Field Declination (D)

Main field declination (D)
Contour interval: 2 degrees, red contours positive (east); blue negative (west); green (agonic) zero line.
Mercator Projection.
⊕ : Position of dip poles

Map developed by NOAA/NGDC & CIRES
http://ngdc.noaa.gov/geomag/WMM
Map reviewed by NGA and BGS
Published December 2014

Source: (NGA; DGC, 2015)

3.2.2.5 Route-specific holograms

Relative to the holograms that indicate the suggested route, two kinds of objects are generated. At first, green arrows similar to Fig. 3.2 are placed in each step of the route, one for each node of the graph. These arrows are 3D objects that were modeled using Blender[11]. A different arrow object was designed for each kind of *maneuver* the Google Maps API presents, e.g. an arrow pointing right for *turn-right* and a roundabout figure with an arrow head pointing left for *roundabout-left*. Apart from the model, the description of the current *turn* is displayed on the arrow body. By applying the *Transform.LookAt* Unity directive aimed at its main camera, these objects keep facing the user as to facilitate reading. If a particular node of the graph does not include the maneuver field in its structure, a regular arrow is displayed. By default, these unspecified arrows point to the next arrow object in the route as a means to indicate where the user should go.

The second type of object generated is a regular 2D blue plane parallel to the ground, as it is exposed in Fig. 3.2. These planes are rectangles that link all of the *turns* present in the route together, thus showing a virtual path for the user to follow between

---

[11]https://www.blender.org/

Figure 3.4: User calibrating the application compass according to a hologram.


Source: Authors

arrows. Since not all roads are perfectly straight, the Google Maps API provides a particular graph for each pair of nodes in the route graph. Each of these graphs is a series of coordinates that represent a polyline following the geometry of the real roads, as Fig. 3.5 exemplifies. A polyline is a continuous line composed of one or more line segments (WEBOPEDIA, 2018). In favor of reducing the extension of the response files and since these polyline graphs are usually very populated, they are not transmitted as regular strings containing coordinates, but in an encoded manner. Since the decoding process of these strings is not relevant to the work at hand, it will not be discussed in depth (refer to (AWAN, 2012) for details). The decoding algorithm is present in the *decodePolylinePoints* script and was based on (AWAN, 2012). It suffices to say that, once decoded, these strings present a set of latitude an longitude coordinates. Each of these sets of coordinates are used to create a visual path that resembles the actual shape of the road, including all curves and turns, thus allowing a more precise drawing of the generated route. Even though it would possible to create this detailed route based on each of these polylines, a different approach was taken in order to minimize the number of objects the limited hardware had to process. The navigation API used provides a summary of the entire polyline collection, named the *overview-polyline*. As described by (GOOGLE, 2018), it represents an approximate (smoothed) path of the resulting directions. The overview polyline is a sort of simplified list of points, reducing the number of nodes in its graph along the way. This procedure makes the visual path rougher, in some sense, however it remains virtually close to the

correct shape, while reducing drastically the number of objects that have to be created.

Figure 3.5: Visual example of a decoded polyline.
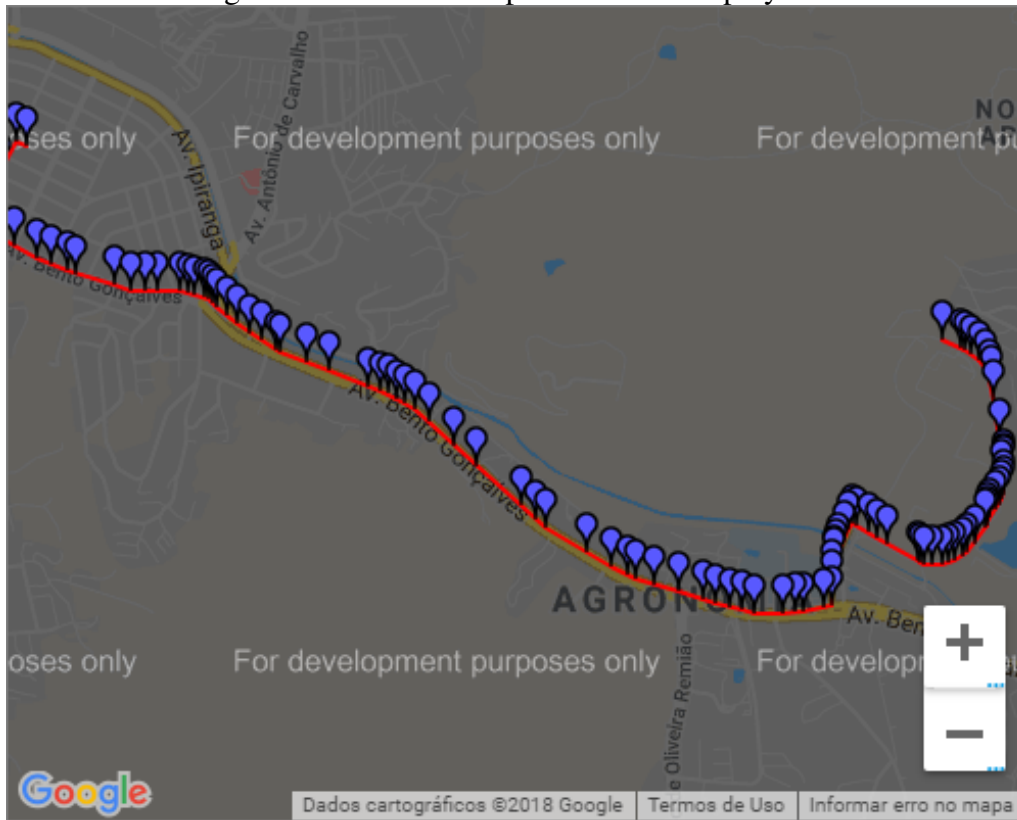
Source: (GOOGLE, 2018)

3.2.2.6 Adjusting objects

Given that not every city is flat, i.e. lacks elevations, an adjustment has to be made in order for the virtual objects to maintain a certain height for visibility purposes. Since the navigation API used is meant for two-dimensional maps, it does not provide altitude data for the extent of a route. Although a specific Elevation API is available, it was not used in order to simplify the task at hand, as the authors did not judge it necessary for the application at its current state to be tested. Thus, an alternative solution was used. After creating a 2D route parallel to the ground, all objects are continuously adjusted in height given the current Z-axis position of the camera, which is a public variable. This translation is smoothed by interpolating the current height of the parent of these objects with the height it should be at, namely a meter below the camera, i.e. a meter below user eye-level. This method is also managed by the *magnetometerController* script, since the specified parent of these holograms is the north arrow object that controls their rotation. This gradual movement eases the visual adjustment for the user, performing the operation

in a seamless manner. Thus, granted that a path may appear to be going directly inside an ascent or off on a tangent in a descent - instead of following their respective curvatures -, the route will still be visible at all times, instead of vanishing out of sight on steep slopes.

Apart from this vertical adjustment, hologram positions are also updated every time the HL enters into a *tracking loss* state, controlled by the *recoverFromTrackingLoss* script, in order to fix any misplacements done by the HMD. Such state is commonly entered when the user looks at surfaces with excessively flat colors in large areas or even when light is scarce, which makes the depth hard to be perceived. When faced with this situation, the application requests a new route from the Google Maps Platform, using however the current GPS location instead of the starting one, resetting all visible path objects. This procedure is also called continuously along the journey, every time the user dislocates in space more than a predetermined distance, based on the chosen navigation mode. Aside from fixing spatial misallocations due to imprecise GPS positions or propagated error, resetting the route also automatically suggests a more relevant route in case the user misses a turn or goes off track due to unforeseen events, such as an accident or a blockade.

3.2.2.7 Additional features

For the sake of improving user experience, a couple of additional features were implemented. Since true MR environments (with head tracking, occlusions, etc.) are still a novelty, interaction approaches in such media have yet to establish a pattern. During our User Interface (UI) preliminary tests, a consensus was reached that placing fixed HUD elements on-screen is not quite natural, apart from wasting a considerable part of both HoloLens small screens. Thus, the idea of granting the user with a mini map to keep track of their journey had to be adapted into a more immersive UX. By using the *trackedHands* state provided by the *UnityEngine.XR.WSA.Input* library, it was possible to track the user hand position and place a hologram object attached to it. This behaviour is managed by the *handTracker* script and was based on (LOZADA, 2016). Thus, whenever users look at one of their hands, a map of the current location is displayed overlapping it, as in Fig. 3.6. This UI allows them to perceive some information about the route, such as what is around them, the distance until the next turn point along with its related instruction and how long is the route regarding both its distance and time. The map erases the information related to the current turn point once the user moves a few meters past its position. It only displays the next one after the user moves halfway towards it, i.e., half the distance between these

two turn points. The map feature also provides users with an easy method to enable an unobstructed view, since by simply putting down their hands, the map disappears out of the HL screen and, thus, out of sight. In addition, performing a tap hand gesture with the map settles its position according to the camera, letting users decide whether they want to constantly see it or not. Tapping again resets the map onto the user's hand.

Figure 3.6: UI map superimposing the user's hand.
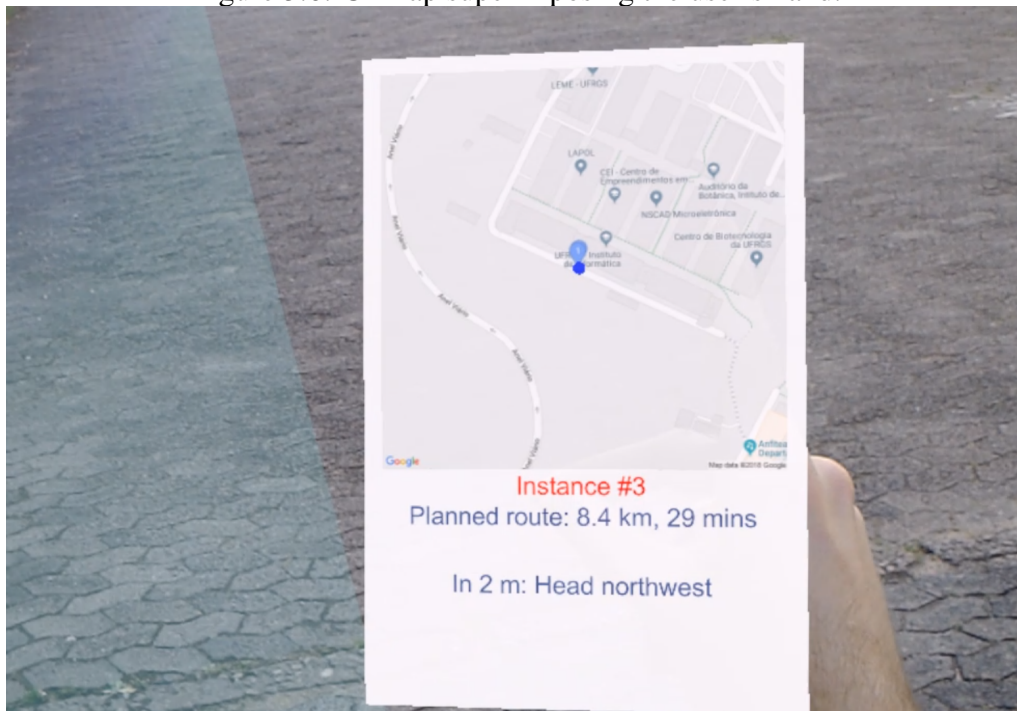
Source: Authors

Navigation may differ based on which vehicle the user is using, given that certain paths should only be done by a specific vehicle. Neither it is safe to walk on highways nor driving on the sidewalk, apart from usually not being legal. Cycling on both sidewalks and highways is also commonly not recommended by law-enforcement. With these points in mind, the API used supplies developers with navigation options, such as driving, cycling and walking. Thus, 3D button holograms were created for users to choose which type of route they desire, as Figure 3.7 shows. These buttons appear along with the calibration UI and are clickable performing the tap gesture native to the HL.

The present section exposed the concept, design and development processes of two applications, namely the holoServer and the holoNav. Their fundamental concepts and methods were presented. Next section intends on exposing the experimental procedures performed in order to enhance the system performance and user experience. Tests with subjects are also discussed, in order to evaluate the system usability.

Figure 3.7: Buttons available for user input.

Source: Authors

3.3 Experimental evaluation

This section aims to disclose some procedures done by the authors in order to test and improve the system, regarding both precision and user experience. The outcome of such tests will be displayed and discussed. Additionally, the protocol intended to be followed in usability tests with actual subjects will be exposed and deliberated.

3.3.1 Preliminary assessment and system performance

During an early development stage, field tests were performed in both pedestrian and motorized vehicle pathways inside the Federal University of Rio Grande do Sul[12], at *Campus do Vale*. Although not highly used by the general public, these pathways are mostly mapped by the Google Maps Platform. Even though their mapping precision is debatable, these were chosen due to ease of access to the authors during development. There, tests regarding route displacement in both rotation and position, horizontally and vertically, were performed.

---

[12]http://www.ufrgs.br/ufrgs/

3.3.1.1 Compass rotation

Rotation tests helped in aligning the magnetometer readings with Earth's true heading. Such tests were somewhat imprecise, due to the unavailability of more correct ways to estimate the north direction than an actual compass and a deviation chart such as Figure 3.3, so these values had to be trusted. Three different Android devices were used to check the stability of their heading values. For this purpose, their gyroscopes were initially reset according to manufacturer instructions. Then, each device, one at a time, was placed atop a wooden table, as to avoid interference from metallic materials in the local magnetic field. Devices were rotated until their heading indicated zero degrees, which theoretically meant they were pointing north, and a line was drawn at the table in order to serve as its default setting. Later, several installments of the following procedure were executed in each test: a magnet was taken and moved close to the device, affecting its magnetic field and, consequently, its heading. After taking the magnet away, interleaved turns of readings were then taken, half resetting the device's gyroscope and placing the cellphone back into position, the other half leaving it untouched. The intention was to check whether resetting it affected precision and which devices were more precise or had more constant readings. Devices tested were a Samsung Galaxy Tab S2 (S2), a Motorola Moto G 1st generation (G1) and a Samsung Galaxy Note 9 (N9). Results are available in Figure 3.8.

An average error of 9.81° was found for the tests where the gyroscope was reset, while the corresponding value of 20.08° represented the remaining tests. In this fashion, resetting the gyroscope during experiments was defined as a good practice. Particular to each device, the S2 presented an average error of 16.07°, the G1 of 4.5° and the N9 27.15°. Thus, although presenting the most precise readings, the S2 presented two error values above 90°, which could be considered critical, given the application. With this in mind, the G1 device was selected for the following tests, since it did not present any critical errors and had the lowest average.

3.3.1.2 Route position

Position tests have demonstrated to be rather unreliable. As has been mentioned, a substantial error is expected when moving great distances through a path. In order to compensate for this, routes are recalculated and updated constantly during a journey, based on distance traveled in any direction, currently every five meters. The current problem,

Figure 3.8: Magnetometer test readings.
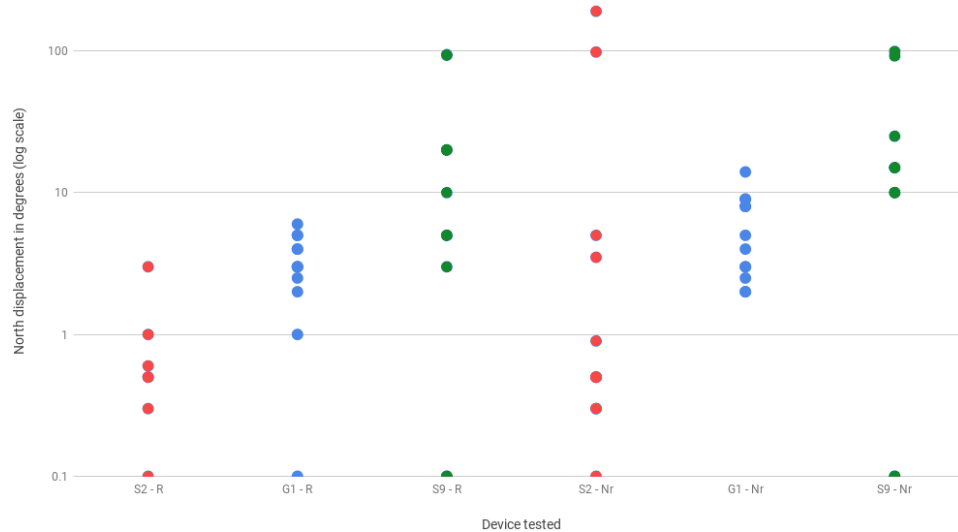
Source: Authors

R marks readings taken after a gyroscope reset, while **Nr** marks readings without resetting the device. **S2**, **G1** and **N9** are three different Android devices. Displacement is expressed as angles measured in degrees. The chart uses a logarithmic scale.

however, is in smaller distances. Simply requesting, mapping and displaying a route in the space around the user already exposes some position errors. Either by GPS tracking imprecision, mapping misplacement of roads or inaccurate approximation values when calculating great-circle distances, tests have shown inconsistent diversions in hologram positioning. These errors are not constant. They diverge in both distance and direction, actually displaying correct locations at times, diverging at approximately two meters at most between their correct location center and the calculated one. In this sense, the reason for this misplacement is inconclusive. Among the possible reasons lie the mapping from geo-located coordinates onto two-dimensional distances. This could be improved by using more precise approximations, however they are indeed quite costly and would slow down the already burdened processing. Given the project at hand and the hardware available, a project decision was made to proceed with tests as is, since the errors displayed may be small enough as not to interfere with user experience.

3.3.1.3 Location updates

In regard to the Android location service, status updates seemed to be too inconstant at first, given the time stamp attribute that is based on a default accuracy parameter

of 10 meters. By reducing it to ten centimeters, updates started to be received more often, roughly once every second, even without moving. This enabled a much more precise navigation, especially while walking, since route updates could now be controlled by code, as previously described, as opposed to waiting for unreliable location updates.

3.3.1.4 Tracking loss

Another crucial finding during early tests was the *tracking loss* state. This happens whenever the HoloLens device cannot locate itself in the surrounding world. Since it uses visual recognition, this usually happens if the image being input displays mostly bland colors, i.e., the image does not have enough details or contrast. Being too close to objects, looking at the ground or even being in poorly lit areas may induce it. The HMD does recover from this state by itself. However, it translates the position of objects from where they originally were to places closer to the user. This feature makes sense on small indoor locations, in order to help users find the holograms they were looking at. For the intended use, it is quite counterintuitive. Tests demonstrated that tracking loss happens rather frequently in outdoor areas, since visual references are usually farther away and more bland, e.g. the sky, roads and walls. This leads to a continuous repositioning of the holograms, misplacing them in the world around, horizontally and vertically, which could cause the route to misguide the user. As to address this issue, the default tracking loss recovery was disabled and a new solution was implemented. Instead of trying to reposition the current route holograms, they are all discarded and a new route is requested from the Google Maps web service, using the current GPS location as origin, thus recreating the virtual path for the user at any given context.

3.3.2 User tests

Although an MR interface has intrinsic advantages for egocentric navigation, a user study with our implementation is necessary to properly assess its practical use. Our test design is based on comparing the holoNav with a well established navigation tool. The present goal is to extract usability information from users regarding each of the two conditions, using both quantitative measurements of their performance and qualitative data from questionnaires answered by the subjects after each test. In the effort of accom-

plishing an impartial test, the Google Maps application[13] was chosen, since both applications fetch their route data from the same source. In the same intention, the order of the two tests was alternated for each subject, as to diminish the impact of the previously tested system. Tests were decided to be taken by users on foot, walking on the sidewalk, for the sake of avoiding any possible accidents, since they had to be performed outdoors, following mapped roads.

3.3.2.1 Quantitative measurements

Regarding the quantitative dependent variables, the following were measured in each test: route time, geographic displacement and attention efficiency. Both route time and displacement were measured using a conventional fitness activity mobile application, Strava Training[14], which tracks its user's geo-located route and time while performing outdoor activities, such as running, swimming and cycling. Route is displayed in a map with the estimated path the user realized based on their GPS locations during the route. Attention efficiency was measured by the means of a Brain-Computer Interface (BCI) headset, which reads electroencephalography (EEG) signals and filters them. For this, the NeuroSky MindWave headset[15] was used along with the Effective Learner application[16]. These are able to interpret the neural oscillations of a subject's brain into focused attention data, providing a chart exposing the user concentration efficiency during a period of time. The Effective Learner application splits the readings among 6 levels of attention, from "Most Effective" (the most focused on a particular thought, regardless of the thought) to "Least Effective", the least focused on any thought. It is important to note that the application is not able to interpret whether the user was focused specifically on the route or on anything else. It is possible, though, to correlate the peaks of attention to particular actions made by the user, such as looking at a map or crossing the street. The number of times each user checked the handheld device during the mobile application trial was also counted.

---

[13]https://play.google.com/store/apps/details?id=com.google.android.apps.maps
[14]https://play.google.com/store/apps/details?id=com.strava
[15]http://neurosky.com/biosensors/eeg-sensor/biosensors/
[16]https://play.google.com/store/apps/details?id=com.neurosky.hafiz

3.3.2.2 Qualitative measurements

As for the more subjective attributes, users were asked to answer a survey comprised of three questionnaires for each test. The first two were traditional multidimensional assessment tools, namely the NASA TLX[17] (Task Load Index) and the SUS[18] (System Usability Scale). The former was created in the 1980s, and has become the standard test for measuring subjective workload in applications. The latter is a renowned reliable tool for measuring usability in the sense of the effectiveness, efficiency and user satisfaction of a system. The third set of questions was based on the *Mobile or Projector* questionnaire by (DANCU; FRANJCIC; FJELD, 2014), since their comparison has similarities to ours. It evaluates two navigation applications regarding six criteria: route attention, road and traffic visibility, navigational aid, safety, fun and ease of use. However, the possible answers were adapted from a binary choice between both systems to a five point scale for each, i.e., instead of answering which of the two options was best for each criterion, users could evaluate each system regarding each of the six topics. This change was made in an attempt to minimize bias. All of the applied questionnaires followed the Likert scale (ROBINSON, 2014), which allow users to inform how strongly they agree or disagree with a statement. Subsequently, subjects were offered a space to write any suggestions or complaints regarding either system.

3.3.2.3 Test protocol

The same route of approximately 350 meters was chosen for both tests, depicted as a map in Figure 3.9, regarded as a 4 minute walk by Google Maps. One at a time, users were taken to the starting point, marked on the mentioned map as *Travessa do Carmo, 178 - Cidade Baixa*. Both systems were pre-configured with the route as to not require any previous knowledge from the subjects. Before starting, the aforementioned measuring applications and apparel were set up. Subjects were then given one of the devices to be tested (either the HoloLens running the holoNav application or a conventional smartphone running the Google Maps app) and minimal instructions on how to use each system, being told to follow the path presented to them in the device being used, as well as walking at their regular pace. After they started walking, interactions were kept to a minimum, until they claimed to have reached their destination. In case a subject went out of the path,

---

[17]https://humansystems.arc.nasa.gov/groups/TLX/
[18]https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html

a maximum time of ten minutes (over twice the estimated time) was stipulated, as to abort the test and declare it as a failure. Once the subjects arrived at the destination, they were relieved of their apparel, both the headset and location trackings were turned off and they were asked to answer the survey regarding the system they had just used. After completion, users were taken back to the starting point and the procedure was done again. This next test, however, using the remaining of the two systems.

The current chapter aimed to disclose the methods used during the development of this project, such as the required tools, the design, implementation and test procedures, as well as a general description of the final prototype. Following up, next chapter intends on presenting an objective and unbiased review of the results acquired during the user tests. Additionally, a discussion regarding such results will be presented, in order to extract constructive information from that data.

Figure 3.9: Planned route for user tests.
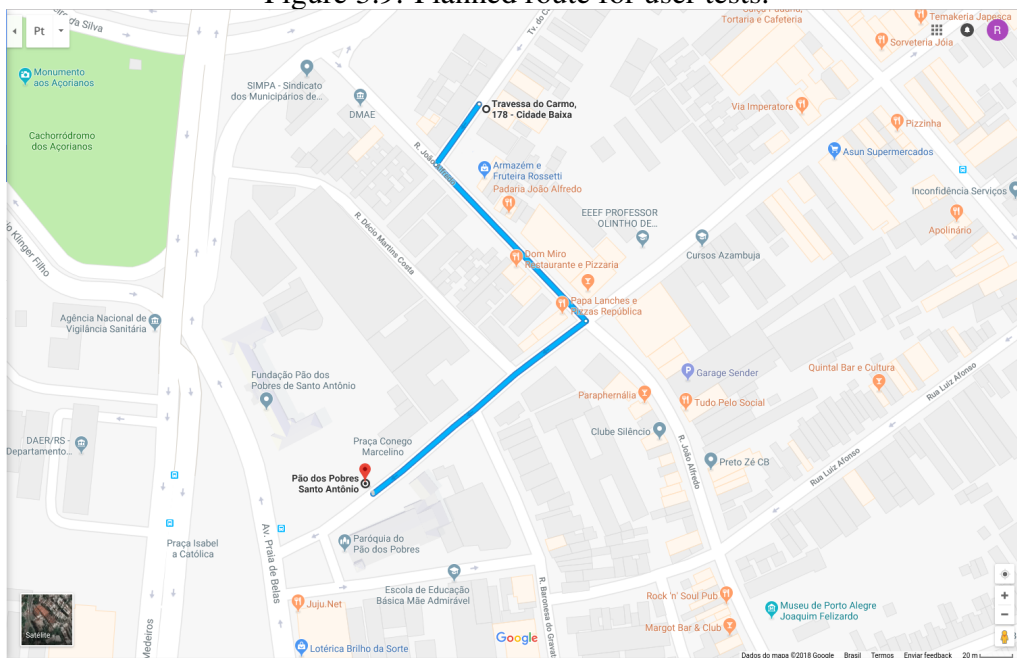
Source: Google Maps

4 RESULTS

The present chapter aims to expose the data gathered from users tests. Firstly, both the qualitative and quantitative results will be displayed, in an objective fashion. Furthermore, an analysis will be made regarding these results, in order to theorize the reasons behind the outcome and how it could impact the future use or further development of the current project.

4.1 User demographics

In total, thirteen subjects participated in user assessments. These were numbered accordingly to the iteration of the test they were part of, from #1 to #13. Subjects were all adults with prior experience in GPS navigation, two being female and eleven male, varying from 21 to 35 years of age, the average age being 26.23 and the standard deviation 3.21. All of them either concluded some type of higher education or are currently near graduation, including four graduate students, in several fields. Out of the thirteen subjects, eight reported to have some type of eye condition, such as myopia, hyperopia and astigmatism. Concerning the most impactful, subject #1 declared to have a scar in the center of his left retina, due to toxoplasmosis as a child, which caused loss of central vision in the left eye, along with an elevated degree of myopia in both eyes, corrected by prescription eyeglasses. Subject #7, although not wearing any type of correction lenses, reported to have an extremely elevated degree of hyperopia in the left eye. Regardless, no effect of eye condition, age or gender was found on the resulting data. Subjects #1, #2, #4, #6, #9 and #11 performed their tests under very bright sun light, which was partially redressed by attaching a piece of dark cellophane sheet to the visor, since the HL does not provide enough brightness to work outdoors. Subject #6 claimed not to be able to see the virtual path due to sunlight being too bright during part of the trial. This highly affected this particular performance, since the application crashed due to overheating of the device, causing the subject not to reach the destination in under 14 minutes. The experiment was then stopped and the route and EEG samples were discarded. Subjects #3, #5, #7, #8 and #10 had problems with the EEG headset coming loose during the tests, which made it either stop recording too early or presented a gap amidst the readings. These EEG samples were also discarded.

4.2 Quantitative results

In order to assess the tests in a more objective manner, measuring tools were used to obtain quantitative data regarding the performance of each individual. The methods for each measurement was described in a previous chapter, while the acquired data is exposed in the following subsections.

4.2.1 Geographic displacement

Fig. 4.1 displays the GPS data regarding the path each test participant trod. All subjects are represented with a different color. Notice that GPS information is somewhat imprecise, which explains some of the trajectories penetrating buildings.

Figure 4.1: Geo-located paths performed by test subjects using the holoNav application.
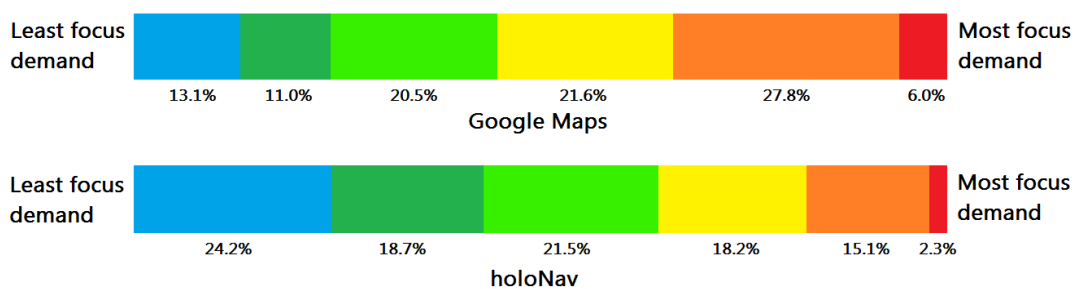

Source: Authors

4.2.2 Time

Comprising all of the valid tests, the average time measured for the experiments with the handheld device was 3 minutes and 57 seconds, with a standard deviation of 30.86. In comparison, the holoNav trials lasted almost a minute more per user, with

an average of 4 minutes and 44.3 seconds. Disregarding the second trial for each user, i.e., accounting only for the tests in which the subject was not familiar with the path beforehand, the respective times were 3 minutes and 42.25 seconds, with a SD in seconds of 31.09, and 4 minutes and 51.33 seconds, with the SD being 59.84. These results expose a roughly 31% increase in time, considering the most unbiased values.

4.2.3 Attention

The number of times users looked away from the route in order to check the navigator application in their handheld device varied from 3 to 14 times. The average being 8.15, with a SD of 3.48. It is important to notice that this test took place in a 350-meter-long route, which means an average of 2.71 glances away from the road every 100 meters. Regarding the EEG results, seven subjects presented no sign of reading errors in both tests. The averages of each level of focused attention measured from these subjects was accounted into the charts present in figure 4.2. Joining together the three higher levels of attention, as the EEG application does, users spent an average of 55.38% of the time at a high level of focus while using the smartphone application. This percentage dropped to 35.64% while using the MR HMD solution. Further discussions regarding these results will be made later in the paper.

Figure 4.2: Subject average percentage of time spent in each focused attention level for each application.

Source: Authors

4.3 Qualitative results

Fig. 4.3 displays the average score for each statement subjects had to respond to. Values indicate how much, in a five point scale, users agreed with each sentence,

regarding the system they had just experimented. Both raw NASA TLX and SUS results are grouped together and indicated by name. The remaining statements represent the third questionnaire used, based on (DANCU; FRANJCIC; FJELD, 2014). The "ease of use" statement was left out since SUS already measures the same quality. The unweighted TLX score was evaluated as 29.81 for the mobile navigator and 39.42 for the HMD application. Meanwhile, SUS final scores were 86.6 for the Google Maps application, dropping to 66.6 regarding the MR solution, respectively an A and a C, according to their standards.

Figure 4.3: Qualitative questionnaire results.
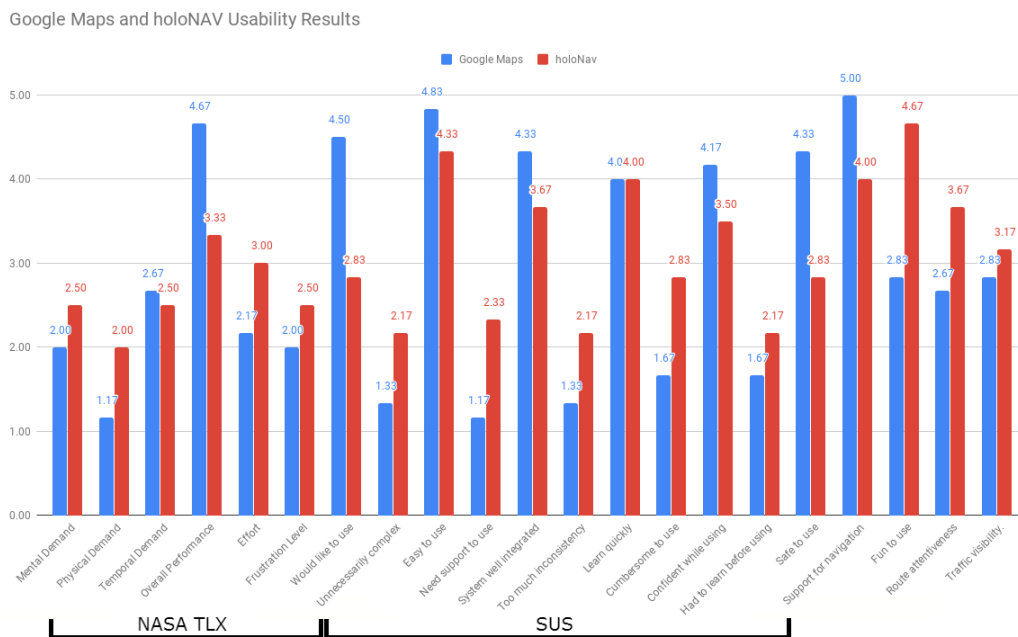
Source: Authors

4.4 Discussion

Most subjects completed the proposed route without any major issues related to the path. Regarding Fig. 4.1, it is noticeable that subject #8, represented in black, contoured the roundabout, instead of taking the shorter path crossing the street and avoiding it. Afterwards, when inquired about it, the user claimed the virtual path at that location did not project a roundabout, merely cutting through it as if it were a regular crossroads, which made it confusing. After the test, it was verified that the Google Maps platform indeed does not characterize that particular part of the map as a roundabout, as shown in Fig. 3.9.

Concerning visibility issues, most users attributed an optimistic higher score to

the proposed MR solution both in "route attentiveness" and in "traffic visibility", when compared to the conventional mobile application. The former attribute aims to categorize the sense of how easy or seamless it was to perceive the suggested route and understand its instructions based on the real world. The latter, on the other hand, concerns how low each application affected the capacity of users to see the oncoming traffic and any important aspects of the road, such as a hole or a pedestrian crossing it. Both categories are highly relevant since the standard use of a GPS navigator reportedly distracts drivers and may cause accidents, as already mentioned. The goal here was to make explicit how much the user can get drawn away from the path while glancing at a device.

Temporal demand was not taken into account during the tests, i.e., subjects were never instructed to complete the route under any amount of time, but at their own pace. Nevertheless, time was measured in order to have a rough estimate about how much the novelty aspect of the system would interfere with user progress. A substantial increase in route time was measured, which could be explained by a variety of reasons. Given the "fun to use" aspect, displayed in Fig. 4.3, it is arguable that users demonstrated a considerably higher interest in the holoNav solution, considering the other to be more ordinary, since many people use it daily. This could represent a crucial explanation for the time increase, given that one is more likely to spend time on something interesting than mundane. Another possibility is that it took more time to understand the basic principle and function of each virtual object. This is more unlikely, however, since the average rating for both easiness to use and quickness to learn were quite high, comparable to the same scores of the handheld application.

Some of the less optimistic SUS ratings, such as inconsistency, complexity and system integration, although not badly scored, could be improved. Refining the geo-location to 2D coordinates mapping algorithms, being able to access the proper HoloLens magnetometer and increasing the screen brightness could potentially improve, if not entirely fix, these problems in a meaningful manner. However, all of these are highly dependable on improving the hardware used, since even performing more precise calculations on the device would be enhanced by a faster processor, which consequently requires more battery, expels more heat and demands a heavier headset, which leads to a final consideration: user comfort.

The least favorable aspects of the proposed solution are largely related to the UX of the HMD *per se*, both in the TLX and SUS tests. Mental and physical demand, effort and inconvenience in using, likeness to use again and street safety are all heavily depen-

dant on the HMD paradigm. Being too heavy, tight and hot, having a screen size too small and even making the user lose balance or becoming nauseous are all common complaints regarding VR and AR wearable devices. It is possible that, as many prominent technologies did in the past, it eventually gains user trust and becomes a more popular device, for either reaching user expectations or people getting accustomed to its flaws. Nonetheless, as user demand increases and the industry adjusts to it, it is customary for devices to get smaller, faster and less expensive over time.

Although a small number, the seven subjects that were taken into account presented similar results in the EEG readings. Six of them displayed a significantly higher level of attention during the handheld device trial, when compared to the HMD. Disregarding the odd one out, an increase of 1.94 times was measured, virtually doubling the average attention level of the user. This ratio drops to 1.76 when accounting for the seventh subject, which is still quite high. Arguably, the higher attention level required could mean a rather great cognitive load demand, which would possibly interfere in other attention seeking activities, such as driving or riding a bicycle. The high number of times subjects glanced at their phones could explain peaks in attention, since mentally translating the route seen in a 2D map into the surrounding 3D world requires some level of concentration. However, no correlation could be observed between these two pieces of data, as seen in Fig. 4.4. Despite a couple of outliers, NASA TLX results show some relation to EEG efficiency in Fig. 4.5, validating user responses to a certain degree. EEG results however seem to display an abundant amount of noise, making it hard to argue in favor of any tendencies with a single sample. Several other speculations could be made based on the EEG numbers exposed in this work. Regardless, further studies should be held with a larger population and in different levels of demanding guiding conditions, e.g. driving in city streets, in order to obtain a less biased and more relevant set of data.
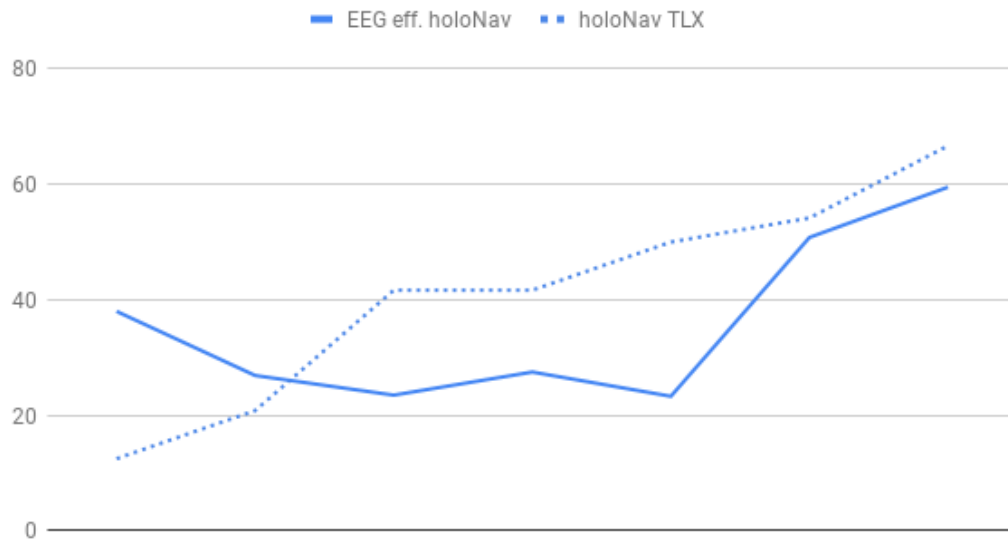
Figure 4.4: Observed number of glances at the handheld device, EEG efficiency and NASA TLX results for each subject.
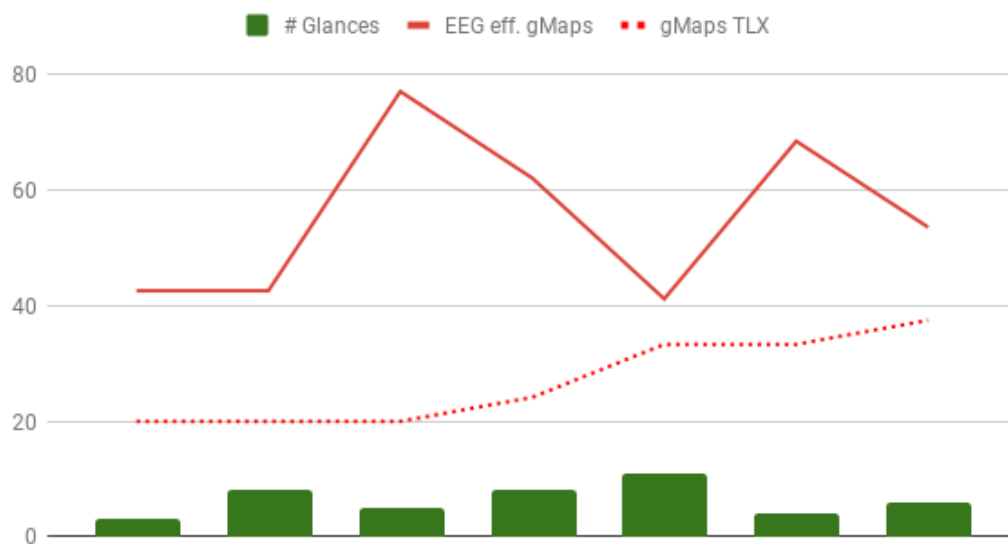

Source: Authors

Figure 4.5: Observed correlations between NASA TLX and EEG results.

Source: Authors

5 CONCLUSION

The present study proposed using mixed reality to support GPS navigational tasks. The conception, design and development of a MR application for the Microsoft Hololens was illustrated, which enables users to request, perceive and follow a geo-located route in the form of virtual holograms guiding them to their destination. Additionally, a usability and performance comparison of it was made against a conventional smartphone navigation application, which is similar to some previous works, such as (DANCU; FRANJCIC; FJELD, 2014). While the most common use of a GPS navigation system is while driving a car, this was not the focus of our research, since primary steps have to be taken prior to testing with such an accident-prone task.

By way of user tests, we demonstrated the performance of the proposed MR application in objectively guiding subjects throughout an entire real-world route. Such an accomplishment provides an interesting advantage, whereas mentally interpreting a two dimensional map into the actual streets surrounding the user is not always trivial, specially in inner-city locations with complex crossroads and a considerable amount of vehicles and pedestrians. This is arguably a difficulty caused by the need to map alocentric orientation (map) to an egocentric task. The wearable MR HMD proved to be a great interface for freeing user attention, mainly visual, since it follows the HUD paradigm. Despite its limitations, it was feasible to build a working prototype capable of day-to-day navigation. This is a highly significant aspect for the current state-of-the-art both in augmented reality and navigation, since new technologies might emerge with more capable and less effort-consuming devices, increasingly more precise, overcoming the current loss of accuracy.

REFERENCES

ALBRECHT, R.; VääNäNEN, R.; LOKKI, T. Guided by music: Pedestrian and cyclist navigation with route and beacon guidance. **Personal Ubiquitous Comput.**, Springer-Verlag, London, UK, UK, v. 20, n. 1, p. 121–145, feb. 2016. ISSN 1617-4909. Available from Internet: <http://dx.doi.org/10.1007/s00779-016-0906-z>.

AWAN, S. **Polyline**. 2012. Accessed: 2018-11-01. Available from Internet: <https://www.codeproject.com/Tips/312248/Google-Maps-Direction-API-V3-Polyline-Decoder>.

BEDING, S. **The Christopher Columbus Encyclopedia**. Palgrave Macmillan UK, 2016. ISBN 9781349125739. Available from Internet: <https://books.google.com.br/books?id=gmmMCwAAQBAJ>.

BROWN, B.; LAURIER, E. The normal natural troubles of driving with gps. In: **Proceedings of the SIGCHI Conference on Human Factors in Computing Systems**. New York, NY, USA: ACM, 2012. (CHI '12), p. 1621–1630. ISBN 978-1-4503-1015-4. Available from Internet: <http://doi.acm.org/10.1145/2207676.2208285>.

BåGLING, M. **Navigating to real life objects in indoor environments using an Augmented Reality headset**. Dissertation (Master) — Umeå University, Umeå, Sweden, 2017. Available from Internet: <http://urn.kb.se/resolve?urn=urn:nbn:se:umu:diva-136490>.

CANADA, N. R. **Magnetic declination**. 2018. Accessed: 2018-11-01. Available from Internet: <http://www.geomag.nrcan.gc.ca/mag_fld/magdec-en.php>.

DANCU, A.; FRANJCIC, Z.; FJELD, M. Smart flashlight: Map navigation using a bike-mounted projector. In: **Proceedings of the SIGCHI Conference on Human Factors in Computing Systems**. New York, NY, USA: ACM, 2014. (CHI '14), p. 3627–3630. ISBN 978-1-4503-2473-1. Available from Internet: <http://doi.acm.org/10.1145/2556288.2557289>.

DEY, S.; KARAHALIOS, K.; FU, W.-T. Getting there and beyond: Incidental learning of spatial knowledge with turn-by-turn directions and location updates in navigation interfaces. In: **Proceedings of the Symposium on Spatial User Interaction**. New York, NY, USA: ACM, 2018. (SUI '18), p. 100–110. ISBN 978-1-4503-5708-1. Available from Internet: <http://doi.acm.org/10.1145/3267782.3267783>.

GARMIN. **Head-Up Display (HUD)**. 2013. Accessed: 2018-10-29. Available from Internet: <https://buy.garmin.com/pt-BR/BR/p/134348>.

GOOGLE. **Developer Guide**. 2018. Accessed: 2018-11-01. Available from Internet: <https://developers.google.com/maps/documentation/directions/intro>.

HOLOLENS, M. **The official Twitter account of Microsoft HoloLens**. 2016. Accessed: 2018-10-31. Available from Internet: <https://twitter.com/hololens/status/811261885450489859>.

HUDWAY. **HUDWAY GLASS**. 2015. Accessed: 2018-10-29. Available from Internet: <https://hudwayglass.com/>.

48

HUDWAY. **Can I use it in daytime?** 2018. Accessed: 2018-10-29.
Available from Internet: <https://hudway.zendesk.com/hc/en-us/articles/
218835697-Can-I-use-it-in-daytime->.

INMAN, J. **Navigation and Nautical Astronomy for the Use of British Seamen**. C.
and J.Rivington, 1835. Available from Internet: <https://books.google.com.br/books?id=
-fUOnQEACAAJ>.

JENSEN, B. S.; SKOV, M. B.; THIRURAVICHANDRAN, N. Studying driver attention
and behaviour for three configurations of gps navigation in real traffic driving. In:
Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.
New York, NY, USA: ACM, 2010. (CHI '10), p. 1271–1280. ISBN 978-1-60558-929-9.
Available from Internet: <http://doi.acm.org/10.1145/1753326.1753517>.

KISS, F. et al. Navigation systems for motorcyclists: Exploring wearable tactile
feedback for route guidance in the real world. In: **Proceedings of the 2018 CHI
Conference on Human Factors in Computing Systems**. New York, NY, USA: ACM,
2018. (CHI '18), p. 617:1–617:7. ISBN 978-1-4503-5620-6. Available from Internet:
<http://doi.acm.org/10.1145/3173574.3174191>.

KUN, A. L. et al. Glancing at personal navigation devices can affect driving:
Experimental results and design implications. In: **Proceedings of the 1st International
Conference on Automotive User Interfaces and Interactive Vehicular Applications**.
New York, NY, USA: ACM, 2009. (AutomotiveUI '09), p. 129–136. ISBN 978-1-60558-
571-0. Available from Internet: <http://doi.acm.org/10.1145/1620509.1620534>.

KUZHAGALIYEV, T. **TCP client in a UWP Unity app on HoloLens**.
2017. Accessed: 2018-10-31. Available from Internet: <https://foxypanda.me/
tcp-client-in-a-uwp-unity-app-on-hololens/>.

LIN, A. Y. et al. Understanding "death by gps": A systematic study of catastrophic
incidents associated with personal navigation technologies. In: **Proceedings of the
2017 CHI Conference on Human Factors in Computing Systems**. New York, NY,
USA: ACM, 2017. (CHI '17), p. 1154–1166. ISBN 978-1-4503-4655-9. Available from
Internet: <http://doi.acm.org/10.1145/3025453.3025737>.

LIU, Y.-C.; WEN, M.-H. Comparison of head-up display (HUD) vs. head-down display
(HDD): driving performance of commercial vehicle operators in taiwan. **International
Journal of Human-Computer Studies**, Elsevier BV, v. 61, n. 5, p. 679–697, nov 2004.
Available from Internet: <https://doi.org/10.1016/j.ijhcs.2004.06.002>.

LOZADA, R. **MICROSOFT HOLOLENS HAND TRACKING**. 2016. Accessed:
2018-11-05. Available from Internet: <https://ritchielozada.com/2016/09/06/
microsoft-hololens-hand-tracking/>.

MAROTO, M. et al. Head-up displays (HUD) in driving. **CoRR**, abs/1803.08383, 2018.
Available from Internet: <http://arxiv.org/abs/1803.08383>.

MEDENICA, Z. et al. Augmented reality vs. street views: A driving simulator study
comparing two emerging navigation aids. In: **Proceedings of the 13th International
Conference on Human Computer Interaction with Mobile Devices and Services**.

New York, NY, USA: ACM, 2011. (MobileHCI '11), p. 265–274. ISBN 978-1-4503-0541-9. Available from Internet: <http://doi.acm.org/10.1145/2037373.2037414>.

MICROSOFT. **Keyboard input in Unity**. 2018. Accessed: 2018-10-31. Available from Internet: <https://docs.microsoft.com/en-us/windows/mixed-reality/keyboard-input-in-unity>.

MILGRAM, P.; COLQUHOUN, H. A taxonomy of real and virtual world display integration. In: **Mixed Reality**. Springer Berlin Heidelberg, 1999. p. 5–30. Available from Internet: <https://doi.org/10.1007/978-3-642-87512-0_1>.

NGA, U. S.; DGC, U. K. **The World Magnetic Model**. 2015. Accessed: 2018-11-01. Available from Internet: <https://www.ngdc.noaa.gov/geomag/WMM/>.

REHMAN, U. **Augmented Reality for Indoor Navigation and Task Guidance**. Dissertation (Master) — University of Waterloo, Waterloo, Ontario, Canada, 2016. Available from Internet: <https://www.researchgate.net/publication/311927005_Augmented_Reality_for_Indoor_Navigation_and_Task_Guidance>.

ROBINSON, J. Likert scale. In: _____. **Encyclopedia of Quality of Life and Well-Being Research**. Dordrecht: Springer Netherlands, 2014. p. 3620–3621. ISBN 978-94-007-0753-5. Available from Internet: <https://doi.org/10.1007/978-94-007-0753-5_1654>.

SYGIC. **Head-up Display**. 2013. Accessed: 2018-10-29. Available from Internet: <https://www.sygic.com/pt-br/truck-gps-navigation/addon/head-up-display>.

WEBOPEDIA. **Polyline**. 2018. Accessed: 2018-11-01. Available from Internet: <https://www.webopedia.com/TERM/P/polyline.html>.

WHIPPLE, J.; ARENSMAN, W.; BOLER, M. S. A public safety application of GPS-enabled smartphones and the android operating system. In: **2009 IEEE International Conference on Systems, Man and Cybernetics**. IEEE, 2009. Available from Internet: <https://doi.org/10.1109/icsmc.2009.5346390>.