Ana Rita de Tróia Salvado

Licenciado em Ciências da
Engenharia Electrotécnica e de Computadores

Augmented Reality Applied to Language Translation

Dissertação para obtenção do Grau de Mestre em
Engenharia Electrotécnica e de Computadores

Orientador : Prof. Dr. José António Barata de Oliveira,
Prof. Auxiliar, Universidade Nova de Lisboa

Júri:

Presidente: Doutor João Paulo Branquinho Pimentão, FCT/UNL

Arguente: Doutor Tiago Oliveira Machado de Figueiredo Cardoso, FCT/UNL

Vogal: Doutor José António Barata de Oliveira, FCT/UNL

FACULDADE DE
CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE NOVA DE LISBOA

September, 2015

Augmented Reality Applied to Language Translation

To my beloved family...

Acknowledgements

"Coming together is a beginning; keeping together is progress; working together is success." - Henry Ford.

Life can only be truly enjoyed when people get together to create and share moments and memories. Greatness can be easily achieved by working together and being supported by others. For this reason, I would like to save a special place in this work to thank people who were there and supported me during all this learning process.

I would like to thank Faculdade de Ciências e Tecnologias from Universidade Nova de Lisboa, for being my second home for the last five years. It was very important to find a place where we have friends and people who care. Arriving early in the morning and listen to the silence, while watching the sea from a distance. Leaving late in the evening, while looking at the stars. Being, not only a student, but also the Student.

A special thank you to professor José Barata. Quoting Bill Gates, "Technology is just a tool. In terms of getting the kids working together and motivating them, the teacher is the most important." It was an honour to work with such a personality, who even introduced me to other bright minds that helped me through this experience. Thank you very much for believing and giving me the opportunity to have a personal touch when choosing the theme, as well as supporting me with knowledge and motivation.

Thank you to all the teachers, who went into my path with expertise and commitment. In particular, Professor Helena Fino: a woman in a men's Department, a character, the Female Character from MIEEC. Thank you very much for assisting and guiding me throughout my journey. For being available every time I needed. Thank you for sharing wisdom, experience and life stories.

A special thank you to my colleagues in the robotics lab, who friendly welcomed me into their working space: Paulo Rodrigues, Ricardo Pombeiro, João Gomes, Francisco Marques, Ricardo Mendonça, André Lourenço and Eduardo Pinto. Thank you so much João Ramalho Carlos, for friendship and companionship. Thank you to my uni's godfather and friend, Jorge Marques Silva, for help and guidance. Also, thank you to my friends, who gave me a life beyond work, full of happiness and joy: Tiago Pereira, Fábio Nogueira, Fábio Lourenço, Bruno Dias, Celso Almeida, Tiago Antunes, Tiago Bento, António Bernardino, Cristiana Nóbrega, António Sá, Sara Ribeiro, Miguel Prego, Gisela Seixas, Rodrigo Francisco, Andreia Ribeiro, and so many others. And dear Mrs. Rosário Grancho, thank you for a smile every day I arrived at the Department.

Ponto Zero, my second family, thank you for bringing gymnastics into my education. For the best moments of fun and craziness. To my coaches, João Martins and Pascoal Furtado, a huge thanks for being in my life.

Finally, thank you so much to my family and close friends: mom, dad, grandpas, godparents... and everyone who has helped and believed in me to achieve so much. Dear Bia, thank you for being with me, even when messing with my patience, but always around with singing, playing, happiness and strength.

Abstract

Being a tourist in a foreign country is an adventure full of memories and experiences, but it can be truly challenging when it comes to communication. Finding yourself in an unknown place, where all the road signs and guidelines have such different characters, may end up in a dead end or with some unexpected results. Then, what if we could use a smartphone to read that restaurant menu? Or even find the right department in a mall? The applications are so many and the market is ready to invest and give opportunities to creative and economic ideas.

The dissertation intends to explore the field of Augmented Reality, while helping the user to enrich his view with information. Giving the ability to look around, detect the text in the surroundings and read its translation in our own dialect, is a great step to overcome language issues. Moreover, using smartphones at anyone's reach, or wearing smartglasses that are even less intrusive, gives a chance to engage a complex matter in a daily routine.

This technology requires flexible, accurate and fast Optical Character Recognition and Translation systems, in an Internet of Things scenery. Quality and precision is a must, yet to be further developed and improved. Entering in a realtime digital data environment, will support great causes and aid the progress and evolution of many intervention areas.

Keywords: Abbyy, Android, Augmented Reality (AR), Optical Character Recognition (OCR), Realtime, Smartglasses, Smartphones, Tesseract, Translation, Vuforia

Resumo

Ser um turista num país estrangeiro é uma aventura cheia de memórias e experiências, mas que pode ser verdadeiramente desafiante em termos de comunicação. Encontrar-se num lugar desconhecido, onde os sinais rodoviários e orientações têm tão diferentes carateres, pode acabar num beco sem saída ou com resultados inesperados. Então, e se pudéssemos usar o smartphone para ler aquele menu do restaurante? Ou até encontrar o local certo num centro comercial? As aplicações são tantas e o mercado está preparado para investir e dar oportunidades a ideias criativas e económicas.

A dissertação pretende explorar o campo da Realidade Aumentada, ajudando o utilizador a enriquecer a sua visão com informação. Permitir olhar em redor, detetar o texto à volta e ler a sua tradução no nosso próprio dialeto, é um grande passo para ultrapassar problemas de comunicação. Utilizar smartphones, ao alcance de qualquer um, ou usar smartglasses que são ainda menos intrusivos, dá a possibilidade de incorporar um assunto complexo na rotina do dia-a-dia.

Esta tecnologia requer sistemas de Reconhecimento Ótico de Carateres e Tradução flexíveis, exatos e rápidos, num cenário de Internet das Coisas. Qualidade e precisão são uma necessidade, ainda por desenvolver e melhorar. Entrar num ambiente de dados digitais em tempo real irá apoiar grandes causas e auxiliar no progresso e evolução de muitas áreas de intervenção.

Palavras-chave: Abbyy, Android, Realidade Aumentada, Reconhecimento Ótico de Carateres, Smartglasses, Smartphones, Tempo Real, Tesseract, Tradução, Vuforia

Contents

List of Figures

List of Tables

Acronyms

ALPAC Automatic Language Processing Advisory Committee

API Application Programming Interface

App Application Software

AR Augmented Reality

ARTrS Augmented Reality TranSlation

BARS Battlefield Augmented Reality System

CDT C/C++ Development Tooling

GPS Global Positioning System

HMD Head Mounted Device

ICR Intelligent Character Recognition

IDE Integrated Development Environment

IMU Inertial Measurement Unit

IoT Internet of Things

JVM Java Virtual Machine

MT Machine Translation

NDK Native Development Kit

NFT Natural Feature Tracking

OCR Optical Character Recognition

PoC Proof of Concept

PRONI Public Records Office of Northern Ireland

QCAR QualComm Augmented Reality

RICS Robotic & Industrial Complex Systems

ROI Region Of Interest

SDK Software Development Kit

UI User Interface

VR Virtual Reality

1

Introduction

1.1 Context and Motivation

The world is a wide place, full of ancient culture, where the people living in it feel the need to connect. Communication is necessary for human relations. However, there is no global language yet. There are about 2,000 idioms known on the planet, which can differ from one another through sound, written accent, reading or even completely different written characters.

Travelling is a hobby loved by many, but without proper communication, it may be hard to completely enjoy. Going to another country can be a true challenge when everyone around does not speak the same dialect. Natural acts, such as arriving at the airport and finding the exit, reading the signs in the street to reach a place, going to a restaurant and know what to ask for, communicating with the local people, they all may become overwhelming. These are just a few issues that a foreigner has to deal with, when going to another country. This is particularly relevant for people coming from regions where English, as a second language, is not well developed. For instance, people from Angola, when travelling to non Portuguese speaking countries, usually require to be with someone that speaks English.

Have you ever been to an exquisite foreign restaurant, all excited to try new food, until the waiter arrives with the menu, from which you see so many different options, with no images to connect the unknown names. There is no way you can choose the best dish without making a lot of questions. Sometimes, you even have to bet, as in a Russian roulette, to decide the winning dish. Having an application prepared to capture the unknown text and translating it to your own language would be a big help in situations like the ones referred before.

Nowadays technology is an every minute resource in life. Everything is becoming more connected. Every device, every equipment. Communication via wireless is making the world smaller. Smartphones have become an indispensable object, with many functionalities, used to help in numerous daily tasks. As a result of combining the previous facts, translation applications are a common target for many developers. Smartphones are currently widely disseminated, either in developed or undeveloped countries, which turns translation applications even more needed. Gadgets like smartglasses have been developed and launched into testing environments in order to keep up with modern and progress needs.

An approach to improve the new applications in a modern and smart way is by using Augmented Reality, a technology that makes reality richer and more interesting by merging layers of digital information with a device's camera view ([Cra10]). This enables internet to be integrated within the user's realtime environment. The ability to allow the customer to experience including data in his view perspective, will give him the illusion of entering into this new "Real Data World".

1.2 Goal and Approach

The goal of this work is to allow people to recognize the information in their own language when travelling abroad. For this purpose, the approach would be implementing an application capable of detecting text in the user's surroundings and immediately translate it to another language. Augmented Reality is a mean to achieve this goal in a creative, innovative and futuristic strategy.

The application may be prepared to run on different platforms. Programming for Android and iPhone devices offers better mobility and flexibility to mobile applications, since it allows them to be accessed anywhere, any time. The development for smartphones makes the App accessible for different types of costumers. Smartglasses are also a target widget, so it is possible to explore and experiment the recent Augmented Reality ambition with the most modern equipment.

The project intends to look into techniques of Optical Character Recognition and analyse them when processing a photograph or displaying realtime results. The methods should be compared and tested with various sample images, to evaluate the Proof of Concept application's versatility. A Machine Translator is also considered and implemented, so that an opinion can be built around this theme.

The results have to be suitable for using the PoC application in the real world. It should deal with natural adversities of the surroundings and perform a user friendly program, with fast and accurate output.

Robotic & Industrial Complex Systems (RICS) group is a team of researchers affiliated to Universidade Nova de Lisboa, that explore both Mobile Autonomous Robotics Systems and Industrial & Intelligent Manufacturing Systems, further detailed at `http://rics.uninova.pt/`. This project is a research work with the purpose of exploring the Augmented Reality concept while enhancing the RICS group with knowledge in this field.

"ROBO-PARTNER" intends to lend a safer hand to the human in industrial automation systems. Also, "Self-Learning Production Systems" are being researched and developed to better fit the needs from society. Imagine the advantages of being able to control the ROBO-PARTNER, while seeing information about the whole system dynamic that only a machine would see. Or even controlling an autonomous boat from a distance and seeing where he goes, in order to help its autonomous system in dangerous situations. AR supported by the smartglasses can provide a revolutionary easy control upon these matters.

AR is a new technology rising great expectations, meant to be integrated in several developing projects. Therefore, this dissertation intends to analyse some methodologies to deal with smartglasses and the use of Augmented Reality.

1.3 Dissertation Structure

This dissertation is organized in six chapters, along with this section, and one annex to support the study:

- **Chapter 1: Introduction** presents the work and its approach. The motivations are outlined and the architecture is explained.

- **Chapter 2: State of the Art and Supporting Technologies** shows the history behind the technology. Several interesting considerations are explored, in order to find a background of existent features and analyse the possible approach strategies. The research looks for new ideas and potential income.

- **Chapter 3: Logic Architecture** enrols the PoC basic blocks structure and a description of the schemes.

- **Chapter 4: Implementation** analyses the programming environment and details some extra cautions related to processing the libraries and source code. Moreover, the OCR and Translation methods are described, so the reader may understand the work behind the project.

- **Chapter 5: Results** gets into the application results, as the name implies. The used techniques are compared and tested with 20 different sample images, that cover some processing features. The experience of experimenting the App with the smartglasses is also referred. Then, three similar applications, already available on the market, are tested and compared to the PoC.

- **Chapter 6: Conclusion and Future Work** summarizes the study and its achievements. Further comments, critics and improvements are taken into consideration, so the project may have a thread of evolution and progress.

- **Annex A: ARTrS User Manual** has a small explanation of the user's steps to manage the PoC application.

2

State of the Art and Supporting Technologies

2.1 Augmented Reality

"There are some people who live in a dream world, and there are some who face reality; and then there are those who turn one into the other." - Douglas H. Everett

Augmented Reality (AR) is a technology that permits interaction with the real world through a device, [Oli15], allowing to see further. It is based on connecting images from the real life with a virtual data image. Looking at the building ahead, for instance, and being able to see more information about it in realtime. Or even looking at a person and knowing his/her name and status in a job environment, can be an advantage and, consequently a very used tool.

Virtual Reality (VR) is also a good resource to predict situations. It allows the user to interact with a created virtual world, which should prevent differing what is real from what is not. It provides a sensorial experience to the brain, creating an environment that does not really exist. VR is used mainly for simulators and games.

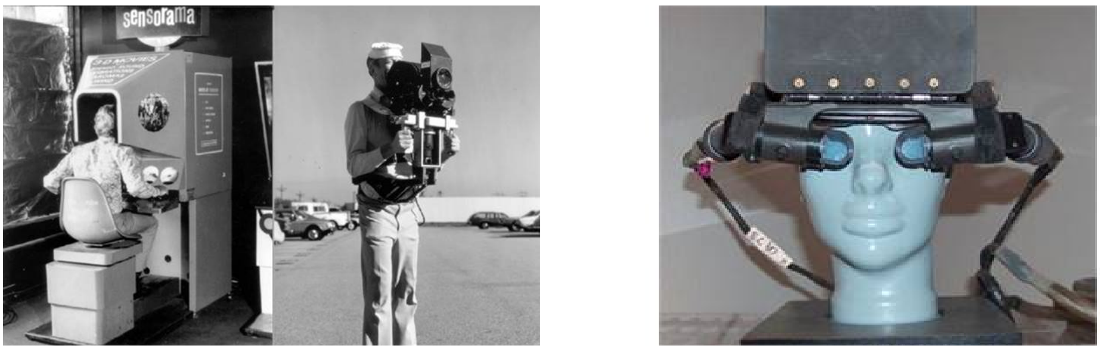
VR is confined to closed areas, since it immerses the user into this completely fabricated world, giving a whole new sense of time and space. Therefore, AR gives many more advantages in daily life tasks, where the user is not completely out of the actual world: keeps the user in touch with the real environment and allows him to interact with the surrounding objects. Both technologies have been released in the 60s and are quickly growing in the market, expecting to continue rising in the future ([McK15]).

Below there are two subsections that introduce a background of AR. They include some historical developments that shaped Augmented Reality through time and a review of three different SDK tools considered for the project.

2.1.1 Augmented Reality: Time-line

Augmented Reality is associated with connecting gadgets and a virtual experience with real senses and the surrounding environment. Back in 1957 Morton L. Heilig, a film-maker, considered the "Father of Virtual Reality", designed and developed Sensorama (figure 2.1(a)), the first machine that allowed the experience with Virtual Reality. It was shaped like 80s' arcade machine and it had a structure around the user's head allowing a 3D stereoscopic projected view. It also provided real sensations such as blowing wind and vibrating the seat [Sun11]. The simulation made the user visit the streets of Brooklyn by bicycle. However, this experience was not appealing enough and it required an expensive budget, because of the filming tactic that made a camera man travel with three attached cameras. The virtual recording was not up to reality. Only in 1961, the technology got the patent. During the 80s, his research and ideas were seen as revolutionary throughout technology.

In 1968, Professor Ivan Sutherland of Harvard University elaborated Head Mounted Device (HMD), the first head-mounted system (figure 2.1(b)), that displayed digital graphics to the user [FN14]. It was suspended from the ceiling of the lab, since it was so heavy for the human head, getting the nickname "The Sword of Damocles" [Sun11]. That time, it was a very futuristic device that started to open people's minds for the benefits of AR [Aic13].



(a) Sensorama. (b) HMD.

Figure 2.1: Technology from the beginning of AR.

Only around 1990, the term "Augmented Reality" started to be used. That year, Professor Tom Caudell brought up a project in neural systems, at Boeing, envisioning the aid on manufacturing in the Aviation Industry. He developed a complex software that replaced the manuals and helped the user on cabling construction [Rea09b]. After being around for some time, mostly on research and within science studies and experiments, people were not aware of the high cost and technology's complexity (both software and hardware). At the end of the 90s, Hirokazu Kato also designed an open source AR Toolkit with video capture tracking and calibration of the camera for any OS platform, enabling the display of 3D objects in the real world [Rea09b].

Since then, the technology became faster, which allowed many developments in the last 15 years. In 1994, Julie Martin created "Dancing in Cyberspace", a theatre show in Australia, where the acrobats and dancers performed and interacted with projected virtual objects in the same environment [Sun11][FN14]. By 1999, the US Naval Research Laboratory started to study Battlefield Augmented Reality System (BARS) to be applied on soldiers training and situation awareness. NASA turns to reliable and low cost spacecraft construction in X-38 program. Bruce Thomas and his team, in the year 2000, created ARQuake (figure 2.2), the first outdoor mobile AR video game.

Figure 2.2: ARQuake technology developed at the University of South Australia.

In 2004, AR was first brought to cell phones by some German researchers [FN14]. Later in 2008 people could really enjoy it on their own phones, thanks to Wikitude AR Travel Guide [Wik15a][joo08]. In 2013, GoogleGlass was released by Google, followed by other companies like Epson with its smartglasses Moverio (section 2.2) and even Innovega with AR in contact lenses experiences.

Augmented Reality is a technology with great potential in a wide range of fields, from medicine to marketing, gaming, military, teaching or even manufacturing, among many others [Rea09a]. Its path is converging with mobile devices, so that it can be completely integrated in people's daily lives (figure 2.3). This year, 2015, Microsoft introduces Windows Holographic and HoloLens [Mic15], with a revolutionary idea of holograms in the real world. The world is changing quickly with the technology. What a few years ago only appeared in movies, is becoming reality. The future remains full of possibilities in the field of AR and the run to evolution has already begun.

There are still some limitations to overcome, such as privacy and excess of information. Some people say that Augmented Reality is bringing even more high tech dependency and people are living more virtual lives than in the real beautiful world. There is a balance somewhere to be found, between living as human beings and yet taking advantage of the quality and experiences that AR can promote.

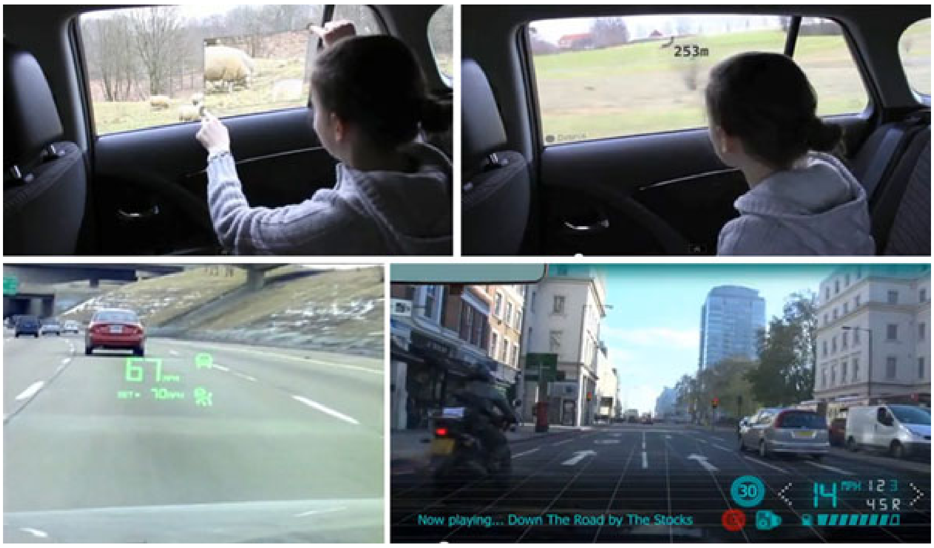
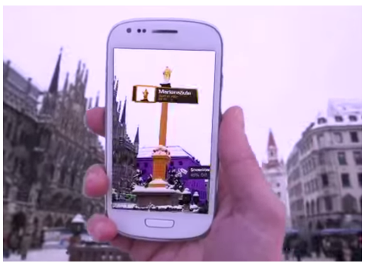


Figure 2.3: AR applied on vehicle industry by Toyota and General Motors [Aic13].

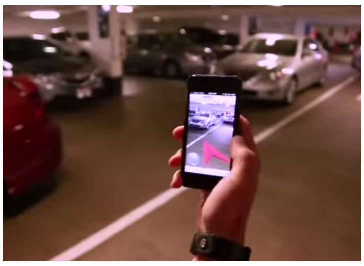
Industries are making their way now through Augmented Reality and the expectations are bright. Alberto Torres, Atheer Labs CEO, points out in an interview [Tay15], that "augmented reality... will transform the global enterprise and the way work is done in the future, in nearly every imaginable way. From the warehouse floor to the operating room, augmented reality will unlock human productivity and enable faster, safer, and smarter workflows for everyone".

2.1.2 Software Development Kit (SDK)

Software Development Kit (SDK) provides tools to help programming some software and usually includes supported documentation from the material. AR SDKs can be structured in categories, such as Geolocation, Marker based and Natural Feature Tracking (NFT) (figures 2.4(a), 2.4(b) and 2.4(c) respectively) [DP15]. The first, allows the integration of Global Positioning System (GPS) and Inertial Measurement Unit (IMU) sensors of the device within an AR application. A marker is represented by a special image to identify and anchor a point in the map [Dev15]. The latter depends on the environment around to create the actual augmented view.



(a) Geo-location based App.



(b) Marker based App.

(c) Natural Tracking based App.

Figure 2.4: SDK organization categories.

The developed SDKs can have different characteristics, that will determine which one of them the developer wants to use. Some of the features are compared in the following table 2.1, [Soc15]. There are several AR engines available online, but only three were considered in this project. They were chosen for comparison via their rates on the Play Store and developer's comments in forums, that highlight the advantages and disadvantages by people's reactions and opinions, in a non commercial way.

Table 2.1: Comparison between some of the SDKs' features in AR, [Soc15].

	3D Object Tracking	Natural Feature	GPS	IMU Sensors	Visual Search	Face Tracking	Content API
Metaio	✓	✓	✓	✓	✓	✓	✓
Qualcomm Vuforia	O (box, cylinder)	✓	✗	✓	✓	✗	O (Vuforia Cloud)
Wikitude	O (expected in 2015)	✓	✓	✓	O (cloud recognition)		✓

One feature to enhance is Tracking, represented in the following table 2.2. This is an important characteristic in an AR App because it is the way to have real interaction between the user, the environment and the device. There are several ways of tracking objects. One of the methods uses sensors, such as GPS, IMU, accelerometer, gyroscope, among others.

Metaio, for instance, does image processing through the camera images. Vuforia distinguishes itself by supporting the recognition of text and continuously tracking the object whether it's visible in the camera or not, called extended tracking. Wikitude provides hybrid tracking, which is the fusion between image recognition, based on NFT, and geo-based properties.

Table 2.2: Comparison based on tracking attributes, [DA15].

	Marker	GPS	IMU	Face	Natural Feature	3D object
Metaio	Id, picture, QR/Barcode	✓	✓	✓	✓	✓
Qualcomm Vuforia	Frame markers, image/text.	✗	✗	✗	✓	✓
Wikitude	Image, Barcode	✓	✓	✗	✓	✗

Many SDKs are available for free to attract and encourage more developers, although they are commonly connected to an additional paid service with many other utilities, as a marketing technique. Hence the licence can also be a way to differentiate the SDK, [DA15]. In table 2.3 it is noticed that the SDKs selected for analysis have this attribute very similar. They all offer free versions with some libraries and databases, and complete paid versions with many more options. None provides an open source code. In this particular case, the cost feature was not relevant enough to decide between the AR engines.

Table 2.3: Comparison based on the license of the SDKs.

	Open Source	Free	Commercial
Metaio	✗	✓	✓
Qualcomm Vuforia	✗	✓	✓
Wikitude	✗	✓	✓

Metaio is known as the world wide leader in augmented technology. It only depends on the device memory to track multiple objects. Even though Metaio seems to be the one to offer more advantages, most of the features are only available as a paid system. Moreover, it has got some limitations with complex 3D objects and the model's size. Vuforia has got its own Cloud Database to store the target images, but there is a limit of 100 images to use. It has got continuous tracking for the objects that go out of view or are placed at great distances. Wikitude uses simple programming languages, such as HTML5, JavaScript and CSS, and allows easy change of platform. Nevertheless it can not track 3D objects, only 2D, and the target object is only recognized with solid colours.

To sum up, there are many existing Apps that use these SDKs and many others to integrate AR features. It's possible to look around and instantly know what the surrounding buildings are [Pri13], evaluate how the furniture looks like in the room [Rid13] or even driving with the display of important information (speed, distance) [New14]. Augmented Reality technology is being highly explored, aiming for an invisible line between reality and virtuality.

2.2 Smart Glasses

"It was summer and moonlight and we had lemonade to drink, and we held the cold glasses in our hands, and Dad read the stereo-newspapers inserted into the special hat you put on your head and which turned the microscopic page in front of the magnifying lens if you blinked three times in succession." - Ray Bradbury, *The Illustrated Man*

Smartglasses are a wearable technology that complement reality with information. They can be hands-free, interacting through speech recognition, or connected to an external device that works with touch commands. As a computer, smartglasses can access the internet and retrieve data from sensors. The input provided is intended to ease mobility, using voice commands, gesture recognition, eye tracking, brain-computer interface, compatible devices and touch screens or buttons.

Many famous companies have started to bet on this technology recently and they see it as the future wear on everyday life, as it will be shown below. They are convinced that this promising gadget is worthy of a large investment in investigation and development, because of the variety of applications that would highly benefit from it and the number of expected interested consumers [Sch14]. Some examples of the existing smartglasses are described above. Google and Epson are two big enterprises developing projects around this theme. Both have already launched their own engines into the smartglasses market, Google Glasses and Moverio, mainly for independent developers to test the new technology.

Google developed GoogleGlasses (figure 2.5(a)), outlined as very lightweight and modern. This technology is designed to work along with speech commands, smartphones and discreet buttons on the hardware [Swi15] and its price rounds €1400. It became available to the developers on February, 2013, but Google took it off sale on 19 January 2015, [Mar15b]. A second version of the device is still expected to arrive this year, according to [All15].

Microsoft has introduced HoloLens (figure 2.5(b)), as previously referred in subsection 2.1.1, and envisions to "make virtual into reality". It is predicted to cost "significantly more" than a gaming console, as a Microsoft executive told the New York Times, [Alv15]. The developer edition is expected to be released in 2016, [Mar15c].

Sony has just started to expand the market of its new SmartEyeglass (figure 2.5(c)) in Japan, Germany, United Kingdom and the US on March 10 [Son]. The price rounds €670 in Europe. Sony states that "has its eyes set on the future of wearable devices and their diversifying use cases, and it hopes to tap into the ingenuity of developers to improve upon the user experience that the SmartEyeglass provides". It also sees "considerable implications for AR, which holds great potential in the domain of professional use as well, such as when giving instructions to workers at a manufacturing site or when transmitting visual information to security officers about a potential breach".

The OculusRift (figure 2.5(d)), from Oculus VR, is also based on some glasses technology and it provides 360 degrees Virtual Reality to the users. It promises to "transform gaming, film, entertainment, communication, and much more" and also "pairs with headphones to make games, virtual worlds and live events feel real". The confirmed release date [Ega15; VR15] is in Q1 2016, for €320.

Epson (figure 2.5(e)), in turn, released Moverio BT-100 in 2012 and now Moverio BT-200 (in subsection 2.2.2) for €640, much light and smaller version. The company states, [Ume14], that "with these improvements, Moverio BT-200 is poised to deliver an AR experience that will revolutionize workflow, training and repair in the enterprise environment".

(a) GoogleGlasses, Google.

(b) HoloLens, Microsoft.

(c) SmartEyeglasses, Sony.

(d) OculusRift, Oculus VR.

(e) Moverio Glasses BT-200, Epson.

Figure 2.5: Smart glasses developed by some famous companies.

There are many shapes for the glasses and prototypes are still being improved to better fit the needs. The interesting areas are plenty, as shown in the subsection 2.2.1 above. The released promoting videos from wearable glasses show the futuristic challenges to be crossed and share many ideas to apply the technology.

2.2.1 Smart Glasses Applications

The number of possible applications for smartglasses is tremendous. There are many fields that can truly benefit with them. The challenge is to integrate Augmented Reality within the Apps. Once this issue is achieved, the sky is the limit. AR can solve all the problems with imagination and innovation.

Education could reach another level of learning with simulators and 3D virtual figures integrating the environment to be studied, where there would be no risk of making, for instance, a dangerous mistake. Driving, flight or surgery simulators are some examples for AR simulation fields. Going to a museum where the characters walk around people would be a very interesting experience. Or getting e-classes where the teacher appears within our sight and he can see what the student is looking at, without being really there, can offer an interactive and dynamic learning method.

Sports could benefit with smartglasses in countless ways. For example, evaluating the information displayed from an athlete's performance or the sport's behaviour, without disturbing the player or the game. The hardware would have to have some adjustments depending on the sport (beach or snow sports), to allow taking pictures, for example, any time without any danger.

As for medicine, useful information displayed alongside the patient, during his examination, would provide a scan of every health levels measurement, with no intrusion of privacy or comfort to the data subject. Smartglasses could have an important role to help blind people, warning them if a collision is predicted or giving directions to a determined landmark.

Documenting ([Sch14]) the most important moments and experiences of life through the automatic camera pictures and videos, or events like natural disasters, where the user may not be able to use his hands, would be a good source of information. Quickly saving events as evidence of crimes could provide more safety too.

Also production in manufacturing ([LK12]) is a possibility with the aid of spacial gesture commands, applied to pick-and-place tasks or motion tracking systems. Having access to information to build equipment, would also save a lot of time and money.

The commerce and marketing would have new ways to display the products and share the news. Lego already released its AR App, Lego Digital Box, where anyone can hold the box of the desired game in front of a screen and watch a little demo of the built puzzle. Ikea has also an App, where the consumer chooses the furniture from its magazine and sees how it looks like when placed in the room, through the smartphone or tablet.

Defence is also a very interested field for smartglasses' technology. In particular, the U.S. Naval researchers have been developing X-6 glasses for the Marines. The head-mounted device supports the warfighters, by allowing them to use weapons in an Augmented Reality awareness situation. According to [Sef15], "the glasses provide information at the speed of light, any time and anywhere. They include a camera with a high frame rate for object tracking and provide an audio capability". A special feature that tells these glasses apart from the others, is their key function to "survive the toughest of environments". A prototype is expected within a year. This sector is aware of the technology developed by other companies, in order to improve as much as possible. It demands high quality and adaptable software and hardware.

Nowadays, F-16 pilots from Portuguese air force already use smartglasses incorporated in their helmets. All the operational terrain data is displayed along side the real view, as well as flight control information. These equipments allow the pilot to drive the aircraft only with his head's movement.

The benefits extracted are irrefutable, however there are issues to be considered [Due14], such as privacy and law, security, social interaction or health disorders. People's discomfort grows as they feel their every step is being monitored: data collection for commercial use or surveillance cameras for security control. As the technology evolves and new unimaginable products keep on coming out, there will be people that do not agree with the change, but in time, the needs will overlap that. With the number of interested users constantly increasing, this technology is predicted to be around for some years and release even more samples on the market, as represented in figure 2.6.

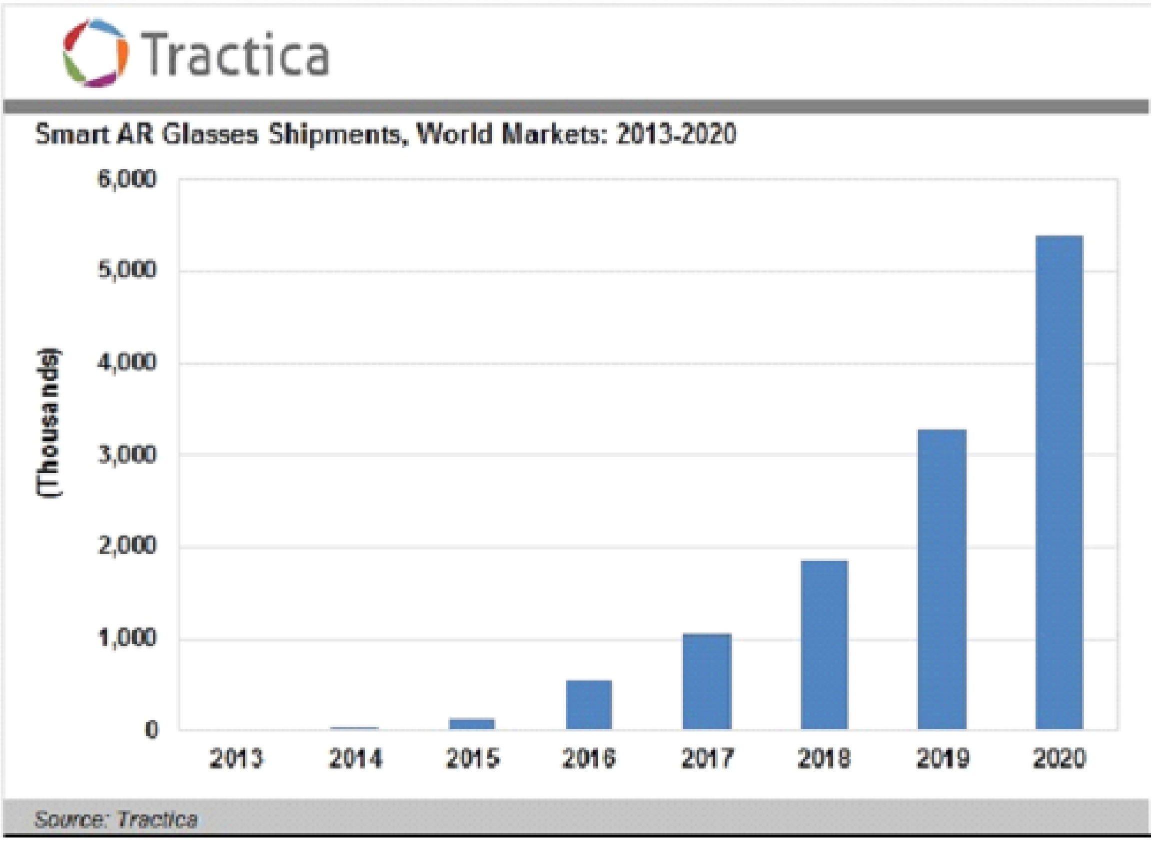


Figure 2.6: Forecast of AR smartglasses in th market, [SP15].

2.2.2 Epson Moverio BT-200

Moverio BT-200 is "Epson's second-generation smartglasses and incorporates much of the feedback provided by both the AR developer and end-user communities", as Anna Jen, director of New Ventures/New Products for Epson America, claims in [Mil14].

The design covers two screens placed in the line of sight, providing digital perspective, and the lens are completely translucent in order to give the user full access to AR. It is not a very fashion product, but still, it fits and adapts the face like normal glasses would, unlike GoogleGlasses [Pra15]. Furthermore, it can easily lay flat on prescription glasses. The device can connect to Wifi and Bluetooth, it has motion sensors and a touchpad, a battery for more than 6 hours and an SD card slot. The included software is for Android 4.0 OS. The AR applications can be supported by the existing front-facing camera. 3D display is also available due to the dual screen of the glasses.

One App already developed for BT-200 is directed to enterprise costumers dealing with maintenance, giving some basic aid to repair industrial equipment (an air-conditioner, for instance), freeing the user's hands to work on it. Another example is applied on warehouses, so that the employees can find the products with more efficiency. Or even recognizing the products in a shelf and evaluate the need to restock.

However, one of the main issues is that BT-200 is not certified by Google so, it can not access Google Play store. In other words, it lacks on Apps that could apply to smart-glasses. The applications can only be downloaded from Moverio APPS, Epson's own market. Besides, the Apps can equally run on smartphones, and since BT-200 is complemented with a control unit device, there is no greater advantage on using it.

Ed English, chief product officer at APX Labs, says that "The Epson Moverio BT-200 is practical, affordable, and powerful enough to handle a wide range of important use cases". The company promises that these glasses can provide an innovative way of building Augmented Reality applications for developers, but there is still some doubt about the real usefulness of the product.

2.3 OCR & Translation Applications

"Our lives will be facilitated by a myriad of adaptive applications running on different devices, with different sensors, all of them collecting tidbits about everything we do, and feeding big digital brains that can adapt applications to our needs simply because they get to know us." - Márcio Cyrillo, Executive Director at CI&T

2.3.1 Existing Mobile Applications

Nowadays, mobile Apps can be a tool for almost every daily activity and they are getting easier to create and share with the world. "App", also known as Application Software, was considered the word of the year in 2010, by the American Dialect Society [Soc11]. With the proper software, common people are able to develop an App for distinct goals, such as games, music, videos, entertainment, health, education, news, among many others. These categories are easily found in any digital distribution platforms (App Store, Google Play, Windows Phone Store and BlackBerry App World [Wik15e]). They provide very little power processing, compared to personal computers and, since they are prepared to operate in any smartphone or tablet, it makes them extremely flexible and portable. Besides, they can quickly interact with integrated features on the hardware, like the camera and numerous sensors.

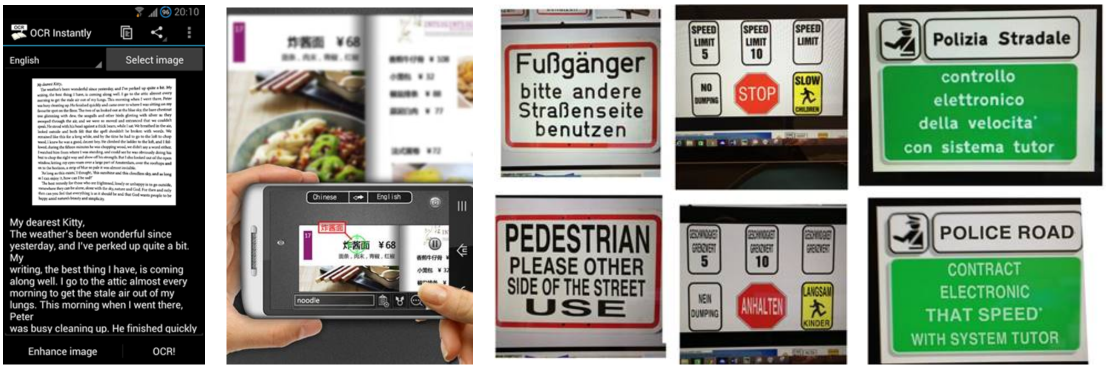
When going to a foreign country, one of the biggest faced challenges is communication. Not only talking with the natives, but also reading the signs to reach a certain place, can be a hard task in hands. When the written characters are totally different between the languages, then, a major issue occurs. That is why the translation software in a mobile device is welcome and can be accessed any time. Even more, to recognize characters in the street, when the alphabet is so different from the available letters on a personal device. This way, with a single photograph or stream video of the word in question, the user is free from typing text.

Several applications already exist inside the fields of Optical Character Recognition and Translation, referred in the following sections 2.4 and 2.5 respectively, and many of them can be associated to AR, described in section 2.1, since they use the camera from a smartphone as intermediate between the reality's view and the digital translated result. Below, some of the translation Apps that exist as a commercial product are referred. They worked as a source of engendering new features in the prototype developed in this project.

OCR Instantly, in figure 2.7(a), is an App offered by TheSimplest.Net, with the purpose of reading text from an image. It is ready to use as free and pro versions, where the latter has more features. OCR Instantly Pro, as a paid service, removes the advertisements, creates *.pdf* files from the selected text, supports 60 different languages to be recognized, provides text to speech, among other features, stated in [The15], its Google Play Store's page. However, the results of the OCR engine targeting some of the languages (Arabic, Hindi, Gujarati, Chinese, Japanese and Korean) are not very efficient. Furthermore, the process to perform the OCR takes several steps: installing the desired languages, taking a picture or getting it from the gallery and cropping/editing the selected image [Gra14]. Although it is not 100% precise, it still offers a good choice to perform text recognition in an image.

CamDictionary, in figure 2.7(b), is described as a "professional instant translator application" in [Int15]. The user just has to point the camera at the text and the translation automatically appears, without the need to take a photo and with little waiting time. There are 36 languages available. The paid service offers text-to-speech and no advertisements. Some believe, [Fre11], this App is not precise enough to get the paid version. Nevertheless, it is a good resource in a foreign country, considered by [Mar13], a "lot quicker than using sites like Google Translate or Babelfish".

WordLens, in figure 2.7(c), is the most similar App for the purpose of this work. It was created by Quest Visual, an American private company, to do free translation in realtime, [Ula15]. The user just points the camera at an area, where the existing text will be quickly translated and placed over the previous language, even without Internet connection. Because of the accomplished bright success, Google acquired it on May 16, 2014 to be integrated inside Google Translate service, which was released on January 14, 2015. Until now, the App is only able to use French, German, Italian, Portuguese, Russian and Spanish languages, although it continues on expanding the available dialects. The creator, Otavio Good believes that "the world around us is very visual, particularly when travelling. There are signs, menus, historical plaques, and a myriad of stores and venues that can leave travellers confused. Word Lens helps translate the world around you simply by overlaying a word-for-word translation of the things you're seeing and reading", referenced in [Wis11]. He also states that AR "has been a neat feature since its introduction. However, Word Lens is a great example of the business opportunities that exist by implementing augmented reality to solve practical problems". Google Translate has been greatly received by the public, as the product lead says in [Tur15]. The technology produces fast and rough translations. It still does not recognize handwritten or stylized text. Too much information on the screen can also be a challenge, in other words, the translation works best for clear printed text, like signs and menus.



(a) OCR Instantly. (b) CamDictionary App. (c) WordLens/GoogleTranslate App.

Figure 2.7: Examples of existing applications.

OCR Instantly is one App example, acquired to the study of this project in order to compare the time and the accuracy of the OCR task. Both CamDictionary and WordLens Apps work for realtime translation. The former requires a paid license to get better accuracy, whereas the latter became completely free, once bought by Google. Overall, these type of Apps are currently being improved by many developers. WordLens brought an admirable imprint to the digital translation Apps' world, with a futuristic concept and a good quick aid for travellers.

2.4 Optical Character Recognition Methods

Symbols are miracles we have recorded into language. - S. Kelley Harrell

I am intrigued with the shapes people choose as their symbols to create a language. There is within all forms a basic structure, an indication of the entire object with a minimum of lines that becomes a symbol. This is common to all languages, all people, all times. - Keith Haring

Optical Character Recognition (OCR) is a technology that detects and recognizes printed text inside an image and converts it into digital format [O.M13]. The group of pixels that represent a letter are compared to the shape of the actual character so that the equivalent can be returned. In contrast, there is also Intelligent Character Recognition (ICR), an engine that enhances the character recognition by reading handwritten text, with a neural network as a self-learning system.

However, having both speed and accuracy can be a true challenge in OCR. When facing the real world, there are many issues to be considered, such as low resolution, picture distortion and rotation, heavy noise or damaged data. Dealing with accuracy can require heavy programming, whereas if the goal is speed, the results can be less precise.

Three case studies are described below. Tesseract, Abbyy and Vuforia enrol different methods, with similar results to the same goal.

2.4.1 Tesseract

Tesseract is a free software used to perform OCR. Created between 1984 and 1994 at HP [Smi07a] and originally coded in C. The code has been migrated to C++ in order to be easily compiled. However, it was never used by HP. Instead, in 2006, Google has acquired the engine and has been developing and improving it. Since then, it became open source, available at [Goo15b], under the Apache Licence 2.0. The last stable released version was Tesseract 3.02, capable of recognizing over 60 languages.

It is also worth mentioning that it was announced that Google Code "will be turning read-only on August 25th", as specified in [Sup15]. Until January 2016, Google Code should work as before, without the links to README.txt files, for example. After the migration of all the projects, the source code will be available under several limitations.

Tesseract can easily detect and recognize black text on a white background or vice-versa, because it implements a step-by-step architecture [Smi07a]. Back then, the techniques applied were considered unusual and it was computationally expensive, but the results have been highly approved.

Nowadays, it is used and recommended by many developers who want to experience OCR engines. Tesseract supports recognition in many languages and simple access to the library, after the costumed installation. Since the user prepares the image according with the caution advices from the software, the results can be very pleasant.

2.4.2 Abbyy

Abbyy developed an Application Programming Interface (API) that performs OCR on images and photographs, through connecting to the internet, accessing a cloud, sending the picture to the OCR server and getting the text results in XML format, as it is stated in [Pro]. It allows free access to do OCR on 100 pages a month and a paid mode for a higher quantity of pages. The free access mode requires registering into Abbyy Cloud OCR SDK console in [Abba], creating an application and getting the password on the email to be used in the code and get the cloud OCR access.

Abbyy Cloud OCR SDK supports three important requirements to perform good Optical Character Recognition: low processing power, works for several mobile platforms and does recognition from low quality images [Sdk15]. It has got 198 different languages available and already integrates trained data.

During FIFA World Cup in Brazil, McDonald's created an application in Germany, where the user had to take a picture to the code printed on the cup or food packaging, as shown in figure 2.8. This code would go to the Abbyy's cloud and directly sent to the lottery database, that would assign a prize to the winner, [Iov15].

Another costumer of Abbyy was Aetopia, a company from United Kingdom. They applied Abbyy's software on the Public Records Office of Northern Ireland (PRONI), which goal is to preserve records of historical, social and cultural character and make them available to the public. The CEO of Aetopia, Aidan McGrath, states in [Sdk13] that

Figure 2.8: McDonald's developed application.

they were looking for a "cloud-based OCR engine to avoid the need to install software in multiple locations" and they "looked at ABBYY based on their reputation for the highest quality OCR. Their Cloud OCR has been a consistent win for us". These statements helped to choose this engine as an object of comparison between the different types of Optical Character Recognition.

2.4.3 Tesseract VS Abbyy

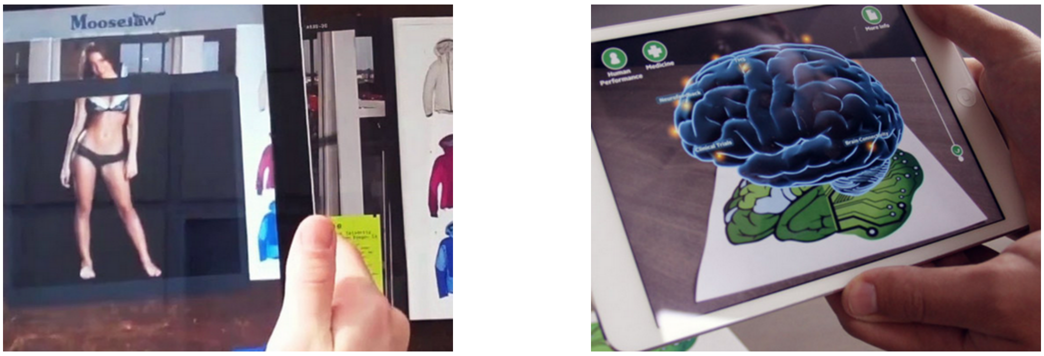
When evaluating the same clean image for both Tesseract and Abbyy, the latter has got some advantage to perform the OCR, because Tesseract's image processing is a little more primitive. However, as Patrick Questembert mentions in [Gro], if the image is managed and corrected, Tesseract is the one that produces better results. Even so, there are some issues to be concerned with, as Patrick enhances, like reconsidering spaces attribution (its elimination or addiction between two letters) and words' mistakes (confusion between VV and W or y and g, for example).

Tesseract software is able to work offline, whereas Abbyy needs internet access to the cloud. However, the requisite of internet connection does not have to be an obstacle in any way, since many believe that everything will be connected to the Internet in the future [Nay]. It seems that everyday more and more people is getting access to the Internet, as [Stab] reports. Furthermore, cloud access is being increasingly used due to its low cost, up to date software, among others, as [sci] refers.

2.4.4 Vuforia

QualComm Augmented Reality (QCAR) Vuforia is a robust SDK with various functions that support Augmented Reality. The technology is a computer-vision based solution and it offers a cloud service to help recognizing and tracking different images. It has an active community to help developers and it is constantly being improved and updated with new versions, such as Unity Extension, Android Native SDK, and iOS Native SDK.

There are several case studies that provide experiencing AR by interacting in the real world through videos and games. For example, Moosejaw X-ray was an application launched by Vuforia that introduces digital media in an innovative way, by allowing to see the catalogue's models in their underwear and choose the clothes to virtually wear (figure 2.9(a)). Another App is Wright State University Brain Scan, developed to visit a 3D brain, as a creative method to educate in the neuromedical area. It was presented in the 2013 Science Olympiad National Tournament (in Dayton, Ohio) to American and Japanese participant students (figure 2.9(b)).



(a) Moosejaw X-ray [LLC14b].

(b) Wright State University Brain Scan [LLC14a].

Figure 2.9: QCAR Vuforia case studies.

A couple years ago, Qualcomm released a new application to do realtime Optical Character Recognition, on Vuforia platform. The software allows the user to detect, recognize and track text in his surrounding environment, through his smartphone or tablet camera. It was meant for education and gaming, detailed in [CM13], as a more advanced and funnier way.

2.5 Translation

"Writers make national literature, while translators make universal literature." - José Saramago

"In good speaking, should not the mind of the speaker know the truth of the matter about which he is to speak?" - Plato

2.5.1 Translation Methods

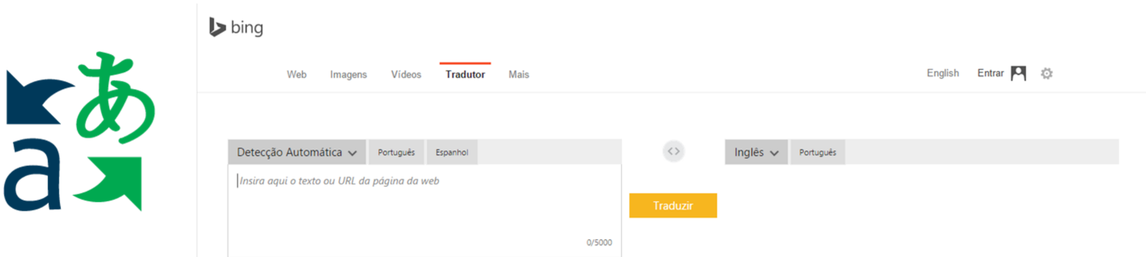
Machine Translation (MT) and Professional Human Translation are two ways of translating services [Goo15a], represented in figure 2.10. The former is very easy to get online, for example, from Google or Bing Translator (figure 2.11), which are both free and that makes them very cost efficient.

(a) Machine translation.

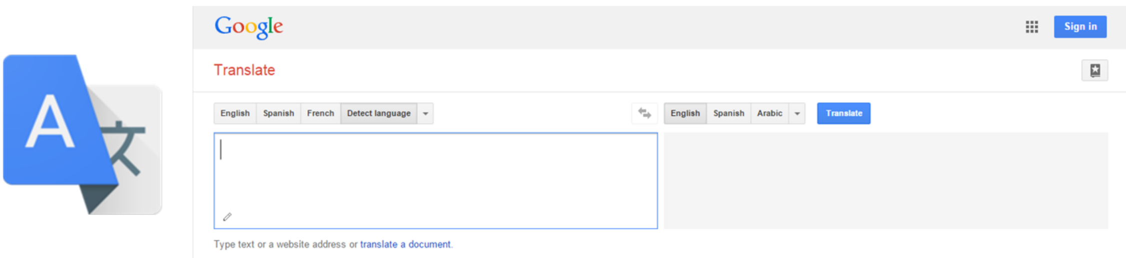
(b) Human translation.

Figure 2.10: Existing types of translation.

However, it is important not to forget that, even though these translators play a big role when translating words and phrases, the software is not intelligent enough to understand the meaning of the whole sentence and the concept of the expression. In other words, the software is not completely independent to compose a coherent translated text [Staa]. Furthermore, the obtained translation hardly gets natural to understand by a native speaker. So if the target goal requires a completely accurate translation, the latter service should be the chosen one.



(a) Bing online translate service.



(b) Google online translate service.

Figure 2.11: Available online and free translators.

This project will look further into Machine Translation, specifically Google and Microsoft Translate. Apart from being the cheapest solution, it also allows very fast and almost instant, translation.

2.5.2 Machine Translation: The Beginning

"It is possible to trace ideas about mechanizing translation processes back to the seventeenth century, but realistic possibilities came only in the 20th century", [Hut05]. The first real opportunity to make this dream come true came up in the 1930s, when Georges Artsrouni and Petr Troyanskii (French-Armenian and Russian) applied for a translating machine's patent. As soon as the first electronic calculators appeared around 1947, computers started to perform a tremendous help for researchers. At 1954, took place the first public demonstration in the United States of America, as a result from the collaboration between IBM and Georgetown Universities. Their project was a very primitive prototype, working with only restricted grammar and vocabulary. The software was being developed by receiving a word as input and giving one or more output translated terms. Meanwhile, some barriers still had to be crossed, like maintaining the semantic of the phrase.

By the year 1966, the Automatic Language Processing Advisory Committee (ALPAC) concluded that this technology was slower, less accurate and also more expensive than hiring a professional human translator. With no brighter results achieved and no progress made, the credibility in MT started to fall. Even so, it was not completely abandoned. Machinery kept on aiding researchers as basic automatic dictionaries. During the 60s, USA and Soviet Union kept on using MT, focused on English-Russian and Russian-English languages, enabling fast translation of technical documents, despite the lack of accuracy. From the 1970s, there was a new purpose for the existence of MT directed to international commerce in Europe, Canada and Japan, which only looked for low cost translations of technical documents.

Only in the 80s, microcomputer-based systems started to emerge from many different countries. This allowed to retake deeper research into MT, looking for a more robust translation around semantic, morphological and syntactic analysis. Microcomputers and text recognition software offered a new and cheaper market for this technology, "exploited in North America and Europe by companies such as ALPS, Weidner, Linguistic Products, and Globalink, and by many Japanese companies, e.g. Sharp, NEC, Oki, Mitsubishi, Sanyo. Other microcomputer-based systems appeared from China, Taiwan, Korea, Eastern Europe, the Soviet Union, etc" [Hut05].

The next decade was revolutionary in this matter. It started to replace the "rule-based", concerning syntactic or semantic rules, for the "example-based" translation system, which dealt with statistical information. It also kick-started speech recognition and translation in various projects throughout the world, such as "ATR (Nara, Japan), the collaborative JANUS project (ATR, Carnegie-Mellon University and the University of Karlsruhe), and in Germany the government-funded Verbmobil project" [Hut05]. Between the late 1990s and early 2000s, MT software's sales have increased significantly for personal use and even more for network services, for example, Alta Vista.

Recently, the main targets for automatic translation are web pages, APIs, videos, files and many other online features that prefer to replace quality for realtime translations. The process has become mainly hybrid [Wik15c], taking advantage of both rule and example based systems. Only few companies keep their main interest in statistical translations, such as Google and Microsoft, that have their own proprietary MT software.

Microsoft Translator software was developed around the year 2000. In 2007, Bing Translator (previous Windows Live translator) came out, with free text and website online translation. Later, in 2011, a cloud-based translate API was launched, becoming available not only for consumers, but also for enterprises [Wik15d].

Google used other online translate engines, like Yahoo! Babel Fish and AOL, incorporated in SYSTRAN software till 2007 [Wik15b]. Then, Google created its own technology with statistic-based translation, [Chi07][Sch07].

These two APIs were developed by world wide known enterprises, Google and Microsoft. Both are very similar in the way they work. The APIs' accuracy and speed are also very close. The former is able to translate up to 80 languages, whereas the latter only translates around 47. Both technologies have come up with an auto-detecting feature that enables the source language to be detected automatically, as the name implies.

Google started to have the Translate API as an open source service in their Google Translate API v1, but it was officially deprecated because of "substantial economic burden caused by extensive abuse", as it is referred in [Pla15b]. It was replaced by Google Translate API v2 on May 26, 2011. Now, as a paid service, Google charges a fee, not only for translating, but also for language detection (€18,31 per 1M characters of text, for each service) [Pla15a]. There is also a default limit of 2M characters per day, which can be increased in the Developers Console. The cost of Google Translate API would be around €36,61 per day.

Microsoft's Bing Translate API is available for free, though it is surrounded by some limitations. The free option requires signing up into the Microsoft Azure Marketplace [Micb] and initiating a new project to get a primary account key and a customer ID, for later use in the code. The free service is provided for only 2M characters per month. To increase this feature, then, a paid service needs to be acquired. The pricing table is explicit at [Mar15a]. For example, requiring 4M characters per month would cost €29,89, whereas with Google API the same service would cost double.

In the end, it is all about the purpose of the project and the budget in hand. These are just two of the numerous tactics on the market. After this analysis, both engines seem to be very similar in speed and accuracy performance. So, the cost of the product was considered the main feature to decide acquiring Microsoft Translator API in this project.

3 Logic Architecture

Logic Architecture chapter outlooks the process design to retrieve text from the real world and instantly translate it. This system covers two main stages: Optical Character Recognition and Translation.

The flowcharts below in this chapter cover the procedure to connect and run OCR (Tesseract, Abbyy and Vuforia) and Translation (Microsoft Translator) techniques that structure the backbone of this project. Each one of the engines approach OCR and Translation in their own ways, either by accessing some libraries or the proper cloud. The overall picture of the logic architecture is represented in figure 3.1.

The two methods that recognize the text from an image (Picture Translation), Tesseract and Abbyy, in section 3.1, lay on OpenCV SDK sample code to control the camera and take a picture of the focused view. This system accesses OpenCV Manager application, which offers an optimized and accelerated performance to process realtime computer vision. The model and architecture are further described in [Ope].

The first OCR method was based on Tesseract's source code, a very recommended system by the programming community. It is displayed in figures 3.2 and 3.3. Several forums suggest this strategy to start managing OCR. It is a free and supported mechanism, that also runs offline. The core architecture is analysed in [Smi07b; Smi07a]. However, because the first testing results experienced some time to process, another method was implemented in order to establish a comparative point of view.

Abbyy is the other technique used in this project to perform OCR on a picture taken by the user, in figures 3.2 and 3.4. It is an online cloud access based procedure that surpasses Tesseract on both speed and accuracy features when dealing with unprocessed images, according to Questembert in [Gro], previously referred. Abbyy's OCR service model is specified in [Abbb].

Vuforia's logic was applied to instantly acquire text in a frame (Frame Translation), in section 3.2, to develop an outlook over realtime Augmented Reality translation. The diagram is represented in figures 3.5 and 3.6. Just like Abbyy, Vuforia is based on online cloud access. Among the available AR SDKs, Vuforia merged the majority of the concepts needed for this AR translation goal: free software, supporting community and realtime text recognition processing. The architecture behind the PoC charts are in-depth explained in [Tra15; Qua15; Lib15d].

The following diagrams resume the main stages of the strategy to achieve the PoC's goal. They describe the Optical Character Recognition process and the Translation activity with their key steps.

A few other features are illustrated, such as language options, connectivity verification, camera and tracker accessibility, among others. The approach changes according to the respective method.

3.1 Picture Translation

Picture Translation is represented in figure 3.2 diagram. The module requires the user to take a picture of the text's area intended for detection and recognition. This section is divided in two different methods, Tesseract and Abbyy, that work in similar ways, but obtain their own results.

Both techniques have the same starting approach. The system displays the camera view and waits for the user input to define the Region Of Interest (ROI). Then, some precautions are attended, in order to avoid unnecessary processing: the program will not proceed to the next stage, unless the internet connection is activated and the source and target languages are different from one another; otherwise, the image is improved for the following actions, by cutting and reducing the desired area and converting it into grey scale.

Above is shown each technique and its strategy to get and translate the text. At the end, unnecessary folders and images are erased from the device.

3.1.1 Tesseract

Tesseract method, in figure 3.3 diagram, starts the OCR operation by correcting the image orientation. Then, the image is sent into the API algorithm, which will return the recognized text.

After that, the Translation operation takes place, with Microsoft Translate API cloud access. The user is able to choose one or more source dialects, which are applied when training the data in the previous stage. For multiple languages, there is an Auto-Detect feature, with a particular use for deciding the source language of the desired text. This is the attribute that distinguishes the two methods. In this case, it may run through the installed language files, or install all the possible files and run through each one of them.

3.1.2 Abbyy

Abbyy method, in figure 3.4 diagram, has the OCR operation surrounded by a timer that controls the amount of seconds spent to access the cloud, in order to interrupt the process when the timer is finished. This procedure avoids wasting the cloud access limits, when getting a probable wrong output - since the more time spent in processing, the more likely the result is incorrect.

Another feature that needs to b verified is the source language. This way, the task can access the Abbyy's software with one or multiple languages, according to the Auto-Detect feature.

Follows the Translation operation where, once again, Microsoft Translate cloud is accessed, so the translated text can be retrieved. If the source language is Auto-Detect the parameters into the cloud request a simple or a complex translation, according to the string with the amount of defined languages.

3.2 Frame Translation

Frame Translation is a module that frees the user from the need to define text areas. The technique processes the camera view, looking for text in realtime. The user should only look at the interested area and wait for the output to appear.

This method was implemented with Vuforia system of Text Recognition, described above. The OCR and Translation used by Vuforia's system is represented in figures 3.5 and 3.6 respectively.

3.2.1 Vuforia

Vuforia method starts by initiating the tracking feature that is always running. Then, the detected text is recognized in the OCR task. The software is continuously running and updating the text found on the screen. The text is only displayed when the last characters are different from the ones previously recognized, in order to prevent constant changing of letters.

The Translation operation is triggered every time the received output is different from the previous one. The same way the other methods do, the detected text is sent into the Microsoft Translate cloud.

The OCR method only recognizes English. Although the source language needs to be English, the Auto-Detect feature can request the translation as an automatic detection of the recognized letters (with no accents) that may create a word from another dialect. As soon as this aspect is settled, the translated text is returned.

3.3 Process Charts

The following flowcharts represent a visual process design that enrols the main steps to perform OCR and Translation for each of the three methods previously referred: Tesseract, Abbyy and Vuforia. These diagrams intend to outline the key modules of the applied logic.

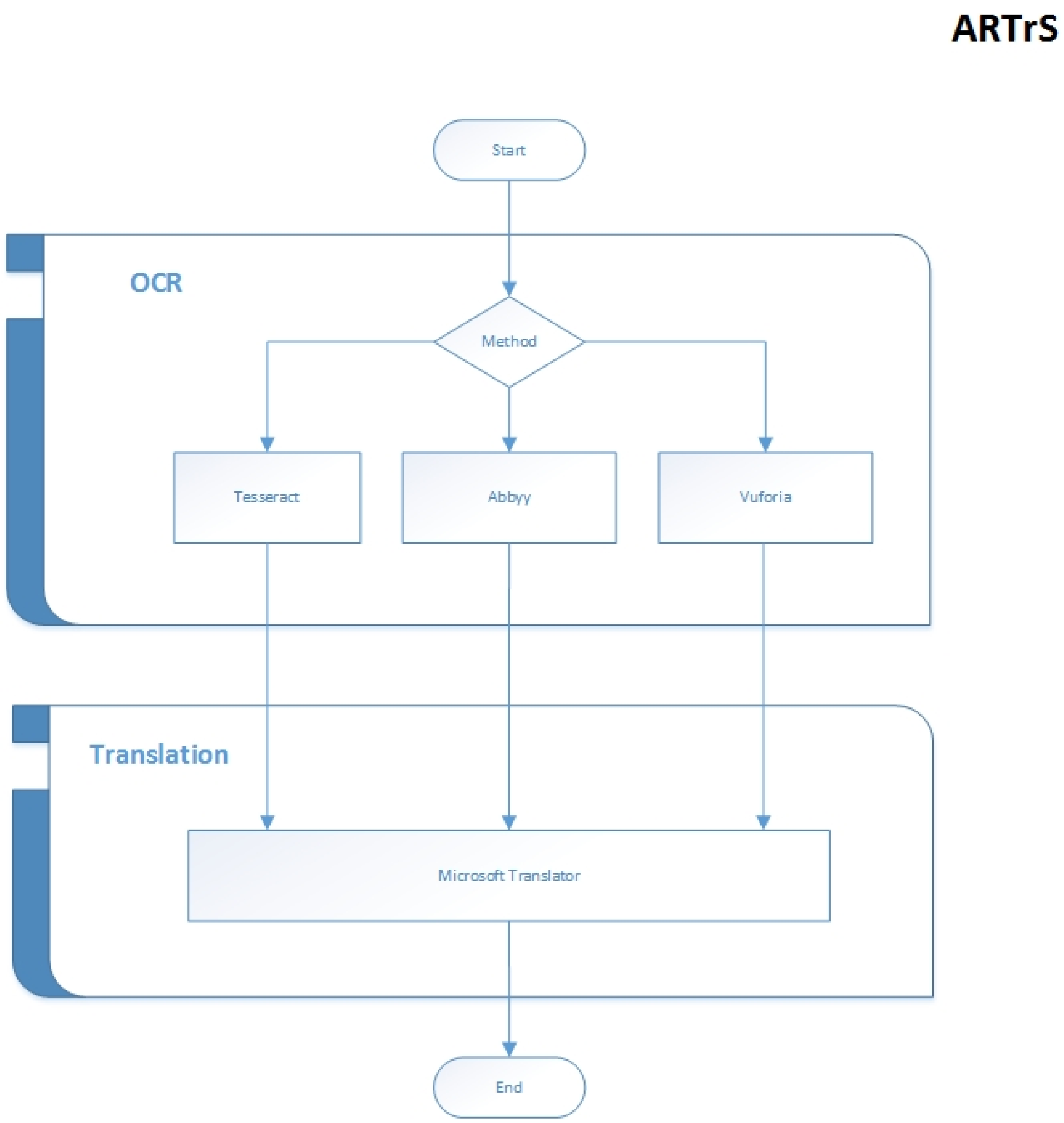


Figure 3.1: ARTrS: Augmented Reality TranSlation - base diagram.

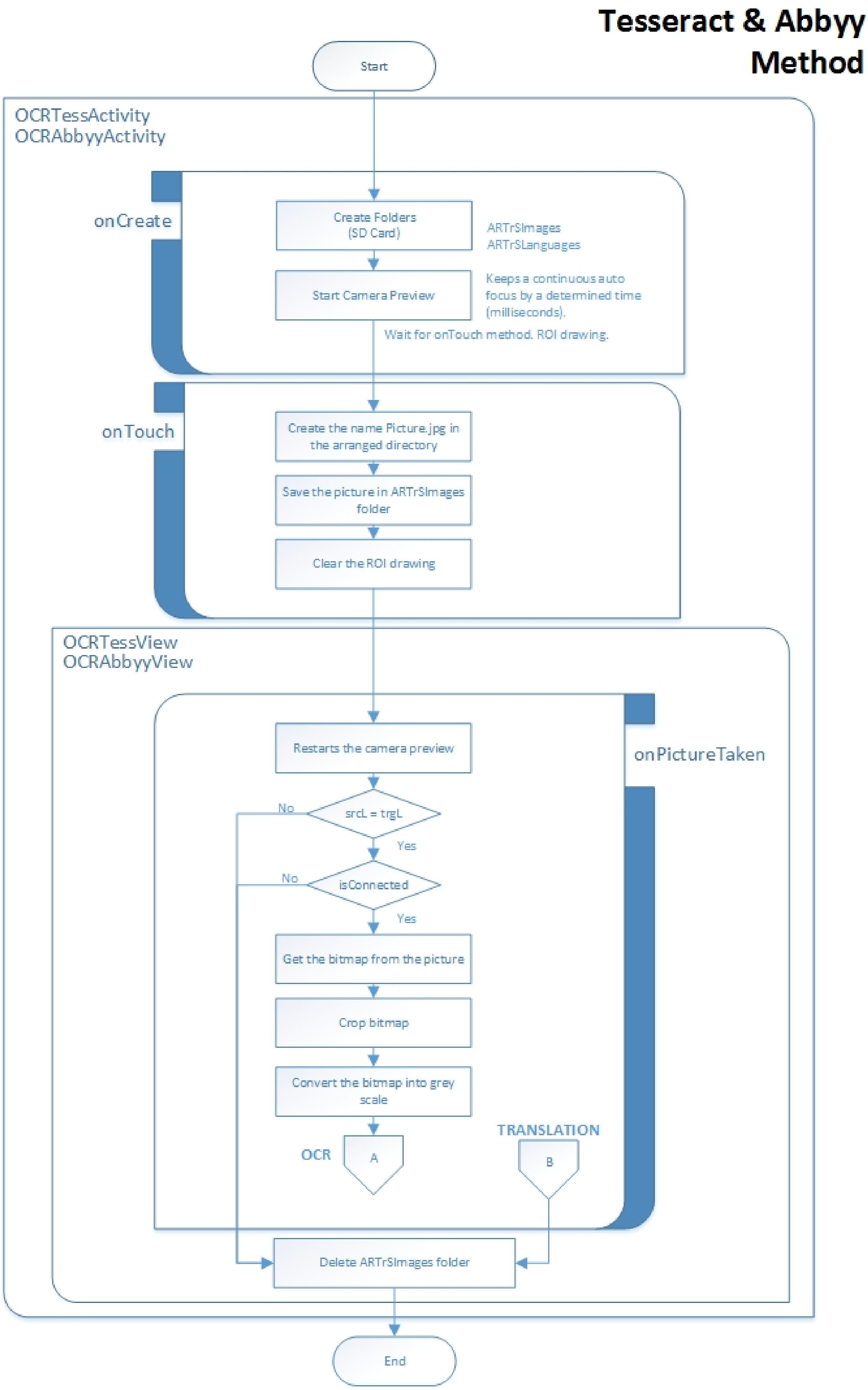


Figure 3.2: Picture Translation: Tesseract and Abbyy - base diagram.

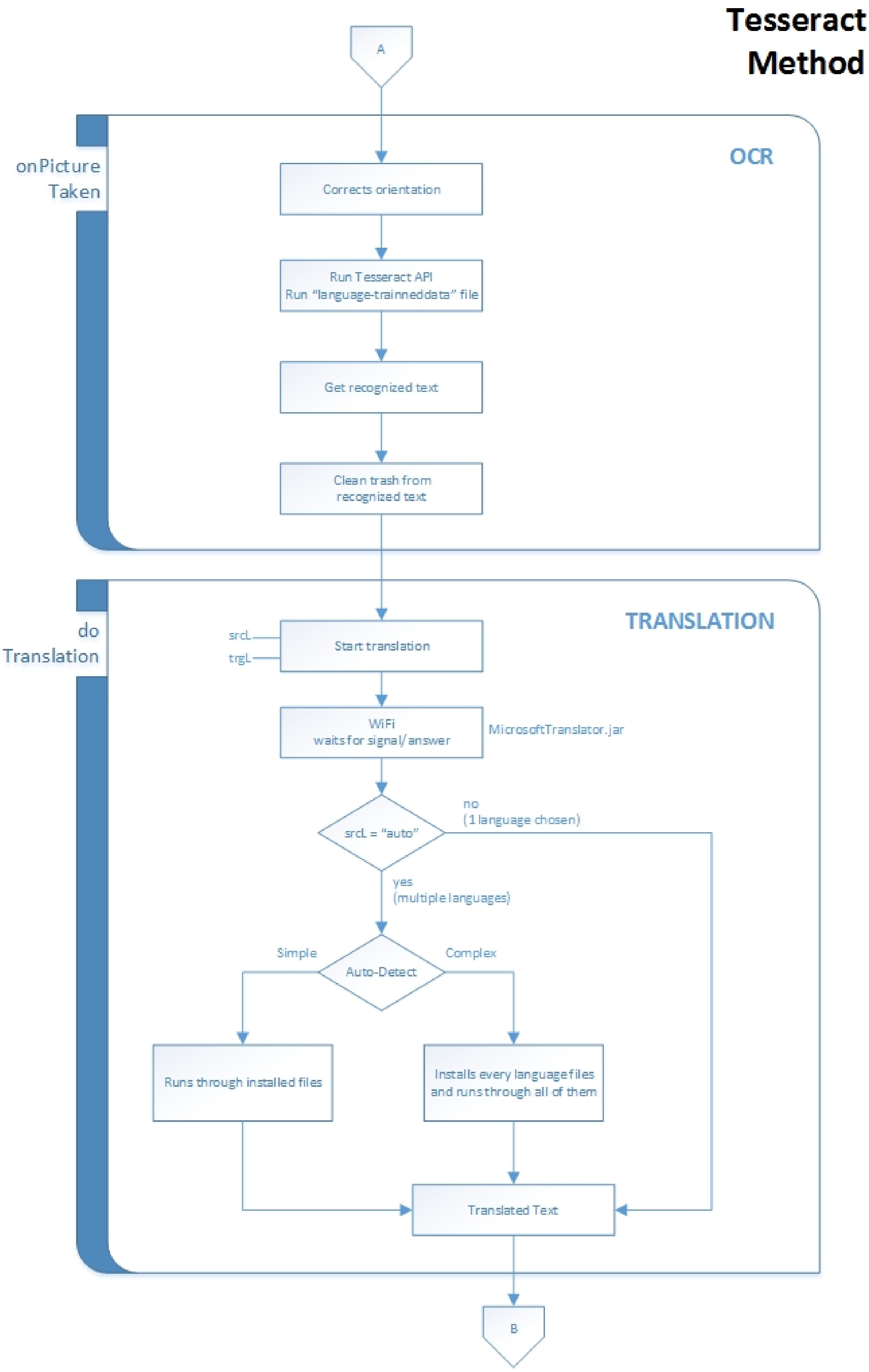


Figure 3.3: Picture Translation: Tesseract - OCR and Translation diagram.

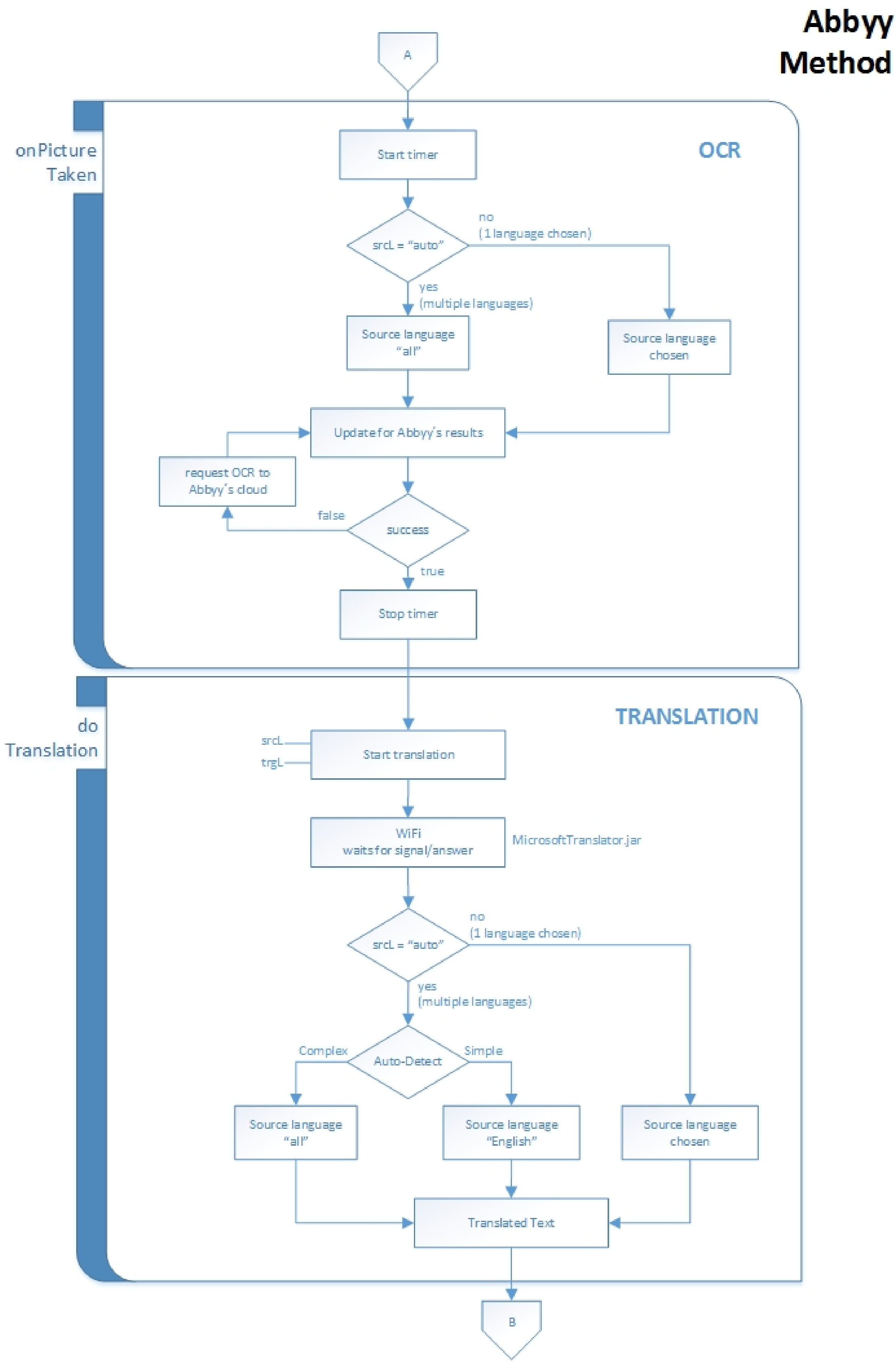


Figure 3.4: Picture Translation: Abbyy - OCR and Translation diagram.

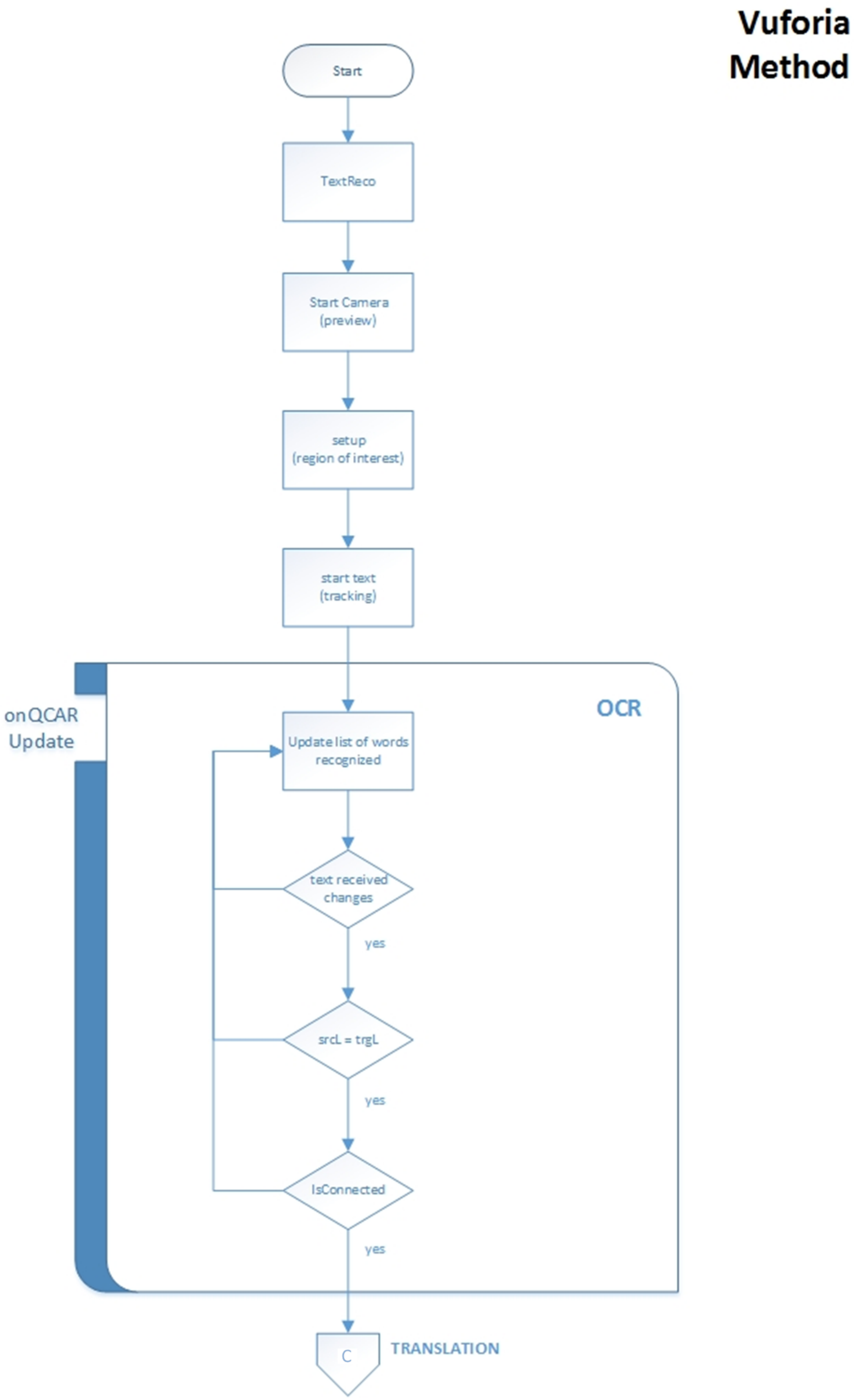


Figure 3.5: Frame Translation: Vuforia - OCR diagram.

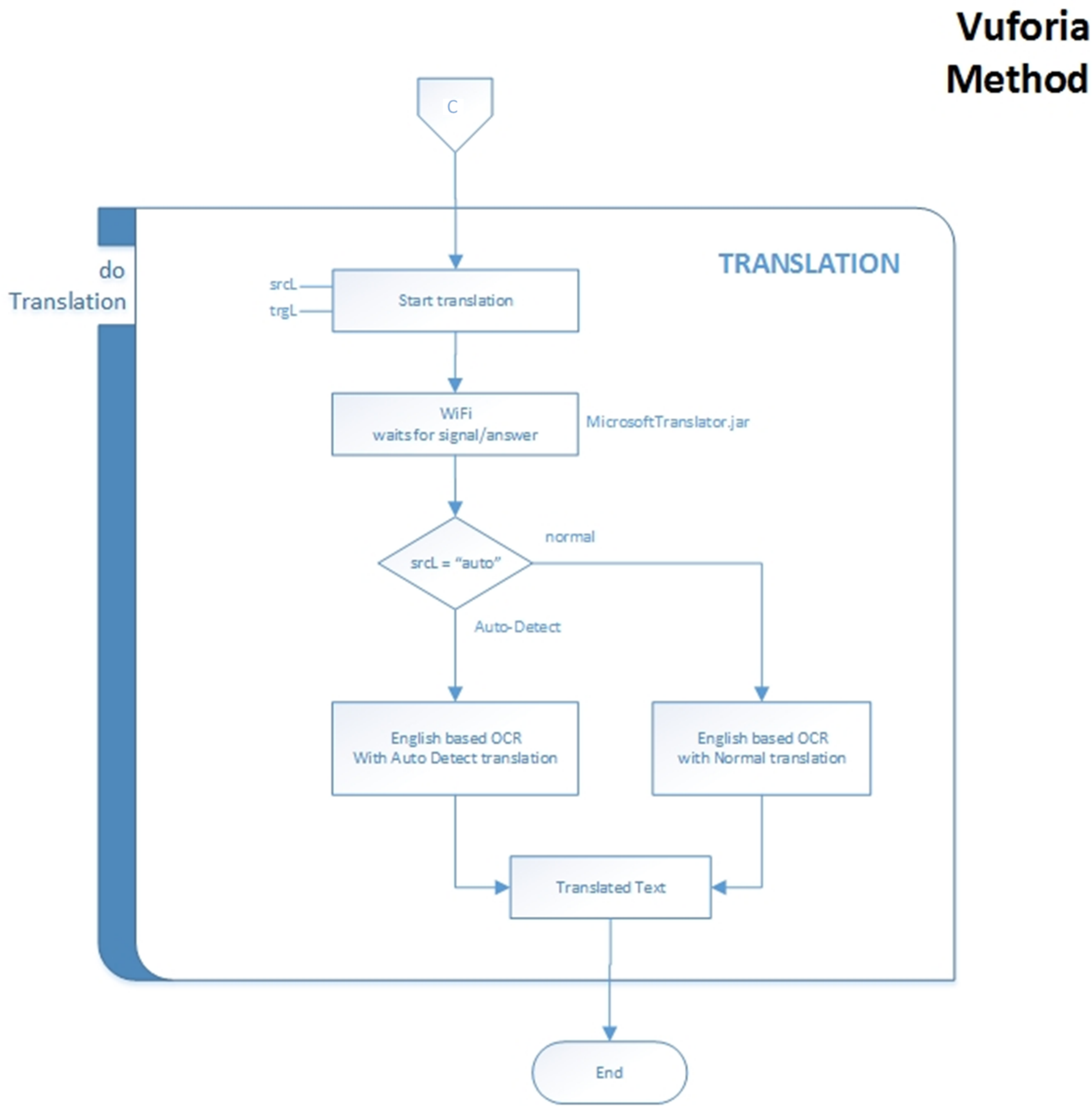


Figure 3.6: Frame Translation: Vuforia - Translation diagram.

4

Implementation

This chapter specifies the methods used in the menus and reports the approach on the demand for the purpose of the work. It starts with a description of the possible environments and the contrast between them, which led to choosing Eclipse IDE for this project. Then, some of the preparations to implement the SDKs and run the application are outlined. After that, the OCR and Translation procedures are explained. The OCR is explored in two different ways, Tesseract and Abbyy methods, which are used in menus 3 and 4 of the application, respectively. Translation is one of the main focus of this work as a complement of information to the real world, and it is used in every menu to return the output asked by the user. The following subsections detail the implemented software and describe its behaviour.

The application Augmented Reality TranSlation (ARTrS) was created in order to apply in real life one of the many purposes and possibilities of Augmented Reality. Being able to get information automatically from the surrounding area is a great step to incorporate a common user into the knowledge of the network, with a simple tap. Therefore, the idea of this App would be allowing a person to make instantaneous translations, without the need for writing the words. As a Proof of Concept (PoC) developed for Android software, it is meant to be implemented and tested on different platforms. The App performs Optical Character Recognition to detect the words in the camera view, by taking a picture of the frame or running in realtime. After that, the output is translated for the user, in the chosen target language.

4.1 Integrated Development Environment (IDE)

4.1.1 Eclipse

Eclipse Integrated Development Environment (IDE) was created by industry leaders Borland, IBM, MERANT, QNX Software Systems, Rational Software, Red Hat, SuSE, TogetherSoft and Webgain. It is an open source environment, designed to run with several languages, although Java is the fundamental coding language, for different platforms. It owns a large and active development community. Additionally, it offers a "sophisticated plugin framework", as referred in [Mue14], that allows the use of other developer tools.

The last available update, to date, was Eclipse Luna. The environment is displayed in figure 4.1. This was the selected working IDE, due to a couple of reasons explained above.

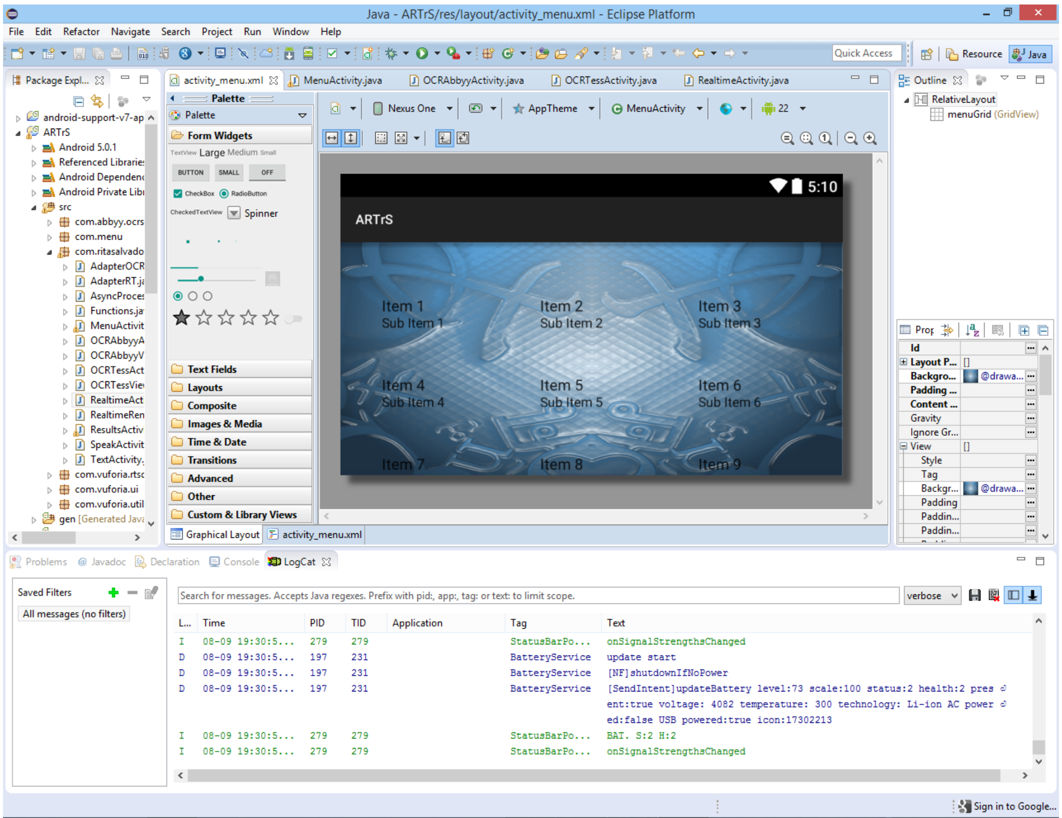


Figure 4.1: Eclipse environment.

4.1.2 Android Studio

Android Studio IDE was created by Google and first released in May 2013. It was designed specially for Android development. This environment has recently become more popular among the Android open source and developing community. It is constantly being updated, in order to become better. The navigation editor, for example, turned out to be very simple and clear for the user. The layout previews' speed performance has improved too. The workflow was updated with multiple shortcuts that eases and accelerates the coding process. With so many improvements, Android Studio has become an environment used by a great number of developers.

4.1.3 Eclipse VS Android Studio

In Eclipse, the user creates a workspace where all the component projects and libraries are located. Android Studio calls these files "Modules" and "Library Modules". Each module has its own Gradle build file with details about the main Android project, as the supporting Android versions and dependencies. The Gradle should be always synchronized with the project. Modules can be run, tested and debugged separately. Both IDEs have similar interface designs that provide component views and interaction with the resources. One of the changes between the two, is the common items, settings and permissions. They went from being manually coded in the Android Manifest in Eclipse, to being automatically added with Android Studio.

Eclipse has been around longer than Android Studio, so its community is much larger. But the latter has highly improved the senior IDE's weaknesses and its development is being regularly updated. It is important to emphasize that Android Studio was built with the purpose of Android programming, whereas Eclipse is a general IDE that works with different languages and platforms.

It is possible to migrate from Eclipse to Android Studio. There are several online tutorials to do so. However, the process is not simple and many issues can occur when exporting and importing the projects.

In conclusion, both IDEs can accomplish the same goals when programming for Android, although Android Studio seems to be the elected option for most programmers. For the purpose of this project, Android Studio requires more computer processing and it is slower to run and compile, while Eclipse seems to be a much more familiar environment.

4.2 Preparation: Additional Cautions

ARTrS application was tested in three different equipments: Samsung Galaxy GT-I9002, Samsung Galaxy Tab SM-T805 and Epson Glasses BT-200. They all operate on Android system. The device should be connected to the IDE for testing. To enable this property, it has to enter into programmer mode by enabling the USB debugger (figure 4.2) in the operating system Developer Options.

The OpenCV library for Android also needs to install an application on the device, in order to access improved algorithms. This App is called OpenCV Manager and it can be downloaded from Google Play marketplace. The application manages OpenCV library with the most appropriate library's version to optimize and accelerate the performance

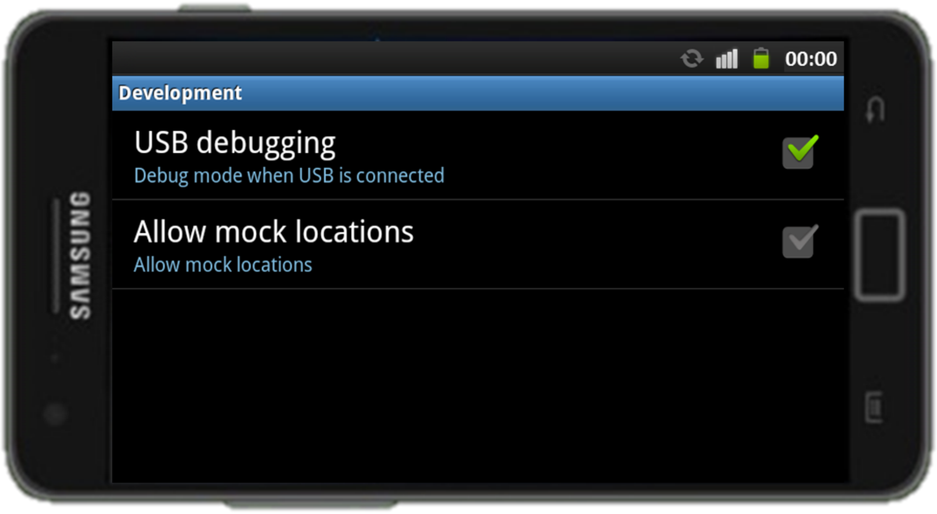


Figure 4.2: USB debugging mode.

of the program. It also provides a lighter and more convenient way to build the project. OpenCV Manager requires at least 2.4.2 OpenCV for Android version to work.

Furthermore, OpenCV runs C and C++ code, while Android applications are programmed in Java. The nature of Java is characterized for running in various hardware platforms, as long as Java Virtual Machine (JVM) is installed. It is portable to run anywhere. In turn, C and C++ needs to recompile the code for all the different hardware platforms. For this reason, C/C++ Development Tooling (CDT) and Native Development Kit (NDK) plugins need to be installed on Eclipse IDE. CDT helps to code C in Eclipse, whereas NDK is needed to compile its source code.

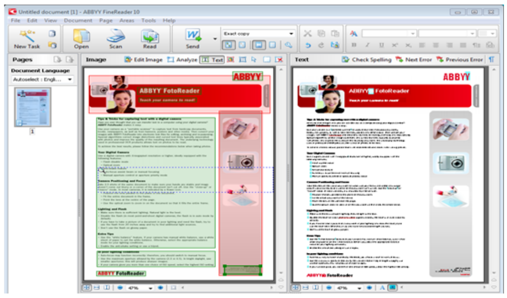
4.3 ARTrS: Augmented Reality TranSlator

The Augmented Reality subject is brought within this program by doing Optical Character Recognition of printed text, through the camera view. The set of letters can be captured from a taken picture or the camera preview frames. The former technique (OCR from a static picture) was implemented in two different ways, Tesseract and Abbyy (figures 4.3(a) and 4.3(b) respectively), whereas the latter was based on Vuforia (figure 4.3(c)) sample code to perform realtime OCR.

4.3.1 Picture Translation: Tesseract and Abbyy

The code to take a picture was supported by a sample tutorial provided by OpenCV 2.4.10 SDK, *tutorial-3-cameracontrol*. It starts the camera preview and waits for a tap from the user, to take a picture of the whole frame and save it in the arranged folder, as a *.jpg* format. OpenCV library provides software designed generally for realtime computer vision.

(a) Tesseract App.



(b) Abbyy software.

(c) Vuforia App.

Figure 4.3: OCR software targeting Augmented Reality.

The application starts by creating the necessary folders in the SD card and waits for the user to draw a rectangle, which should be around the desired text. Then it proceeds to the saving operation, according to the predefined directory. Before doing the OCR procedure, there are some preparations to prevent needless processing. It should be noted that the activities that need the SD Card are only launched after verifying that the card exists. The internet connection is also checked, so that the tasks which are web dependent do not run in vain. Moreover, the taken picture is cut after the rectangle specified by the user, making it smaller for afterwards analysis. Then it is converted into grey scale to improve the following actions. The previous procedures expect to ease the accuracy and speed, when handling Optical Character Recognition.

Subsequently, follows the OCR itself, that aims for detection and recognition of letters and symbols inside an image, as described in section 2.4. Two methods were programmed in this study. Tesseract was the first one to be implemented, but due to its poor results, another mechanism had to be explored. Abbyy came up with a better outcome in terms of both speed and accuracy, heading towards Questembert's opinion in [Gro], already seen in subsection 2.4.3.

Tesseract's OCR method faces an image correction concerning the orientation and, after that, goes through Tesseract API library and its languages files to get the trained results. This technique does not need internet connection.

Abbyy, in turn, needs cloud access to perform OCR. This registration [Con15] grants a specific identification and password that controls and limits the requests of the used space. Since the communication between the cloud and device may not always be successful, the translation asynchronous task in this project is surrounded by a timer, that is running in parallel to avoid extensive waiting time. If the output result returns true, the OCR is successful. Otherwise, the user is notified that something went wrong and he will have to try again.

When the OCR approach is complete, the translation starts. The used tool for this goal was Microsoft Translator API. Whatever the OCR method is, translation follows each one of them in a similar way, with only small adjustments. As a general description, the request sent to Microsoft's cloud needs registration credentials, source and target languages and the text to be translated. If the data is all correct and no problems were detected with the internet connection, the translated words are returned. According to their translation, what differs the three OCR paths is the auto detection feature from the source language. This attribute has different possibilities on each task. Tesseract's auto detection asks the user how many languages he wants to install and run, whereas Abbyy needs to change the string of languages to send to its cloud.

After the translation engine is finished, the images' folder and its content are deleted, so as not to waste any more space on the device with a picture no longer used. Overall, when facing Augmented Reality, the user can translate what he sees, without the need of any writing. Either by taking a picture or simply pointing at the word, the device can get the text read by OCR and return its translation in a matter of seconds.

4.3.1.1 OCR with Tesseract

Tesseract library needs additional support from tools and libraries that have to be included on the IDE. It also requires some files to be installed on the device. There are several online tutorials to run Tesseract on a personal source code. The main supporting tutorial can be found in [Tesa]. However, fitting the process in the project is not simple and it can give many different errors to the programmers along the installation, related, for instance, to native code build tools, such as NDK and Ant. The instruction steps have to be narrowly followed.

The source code in use was developed by Robert Theis and is mostly sponsored by Google. It is free to use and it can be found in [The] as *tess-two* project, with some instructions from the author. It also contains some image processing libraries from Leptonica, an open source site that provides image processing software.

First, *tess-two* project needs to be built apart from the IDE. The following commands were used via Cygwin command-line interface for this purpose:

```
$ cd <project-directory>/tess-two
$ ndk-build
$ android update project --path
$ ant release
```

Then, the project has to be imported into the IDE, in this case, Eclipse. Two errors still need to be noticed and fixed: the project properties should be activated; IsLibrary attribute should be checked to make sure it is read as a library. After this, *tess-two* project must be added as a library of the master project.

After solving the previous issues, the actual code concerning the OCR engine has to be implemented. The picture that was taken goes into the TessBaseAPI to be processed. Listing 4.1 reveals a sample code to perform the communication between the running task and the *tess-two* library's functions that return the OCR of the requested text at the end.

Listing 4.1: Tesseract SDK: Code implementeation.

```
// DATA_PATH = Path to the storage where the picture is saved
// language = for which the language data exists, usually "eng"

TessBaseAPI baseApi = new TessBaseAPI();
baseApi.init("DATA_PATH", "language");
baseApi.setImage(bitmap);
String recognizedText = baseApi.getUTF8Text();
baseApi.end();
```

The language files can be downloaded from [Tesb]. It has to be in mind that, the more languages the application has, the more time it takes to compile, which can be very frustrating when dealing with tests. But once the App is running, it proceeds without any more delay. These files can be easily installed whenever the user needs each one of them, or erased from the location folder. As soon as the detected text is returned, the translation process can start.

Tesseract performs OCR offline, which can be very useful when internet is not available. The configuration is not easy, but once complete, the user is able to recognize a large number of different words. The better environment conditions, the better obtained results.

4.3.1.2 OCR with Abbyy

Abbyy offers Quick Start Guides, code samples and recognize sample images, for programmers to explore and test. The SDK is available for Android, iPhone and Windows Phone platforms, in several languages, such as Java, JavaScript, .NET, PHP, among others mentioned in [Doc15]. It also requires access to the cloud through online registration, as shown in listing 4.2, in order to get the amount of recognized characters.

Listing 4.2: Abbyy API Credentials.

```
Client restClient = new Client();

restClient.applicationId = "AbbyyID";
restClient.password = "AbbyyPW";
```

The algorithm is able to recognize multiple languages, although it is not advisable since it highly declines the performance of the program. If a multi-language feature is needed, than there should not be used more than 5 languages. The more languages are used, the slower the execution gets.

According to the defined source language and the taken picture, an asynchronous task starts requesting the detected characters, listing 4.3.

Listing 4.3: Abbyy SDK: Cloud request implementation.

```
// DATA_PATH = Path to the storage where the picture is saved
// language = for which the language data exists, usually "English"

ProcessingSettings processingSettings = new ProcessingSettings();
processingSettings.setOutputFormat(ProcessingSettings.OutputFormat.txt);
processingSettings.setLanguage(language);
publishProgress("Uploading..");
Task task = restClient.processImage(DATA_PATH, processingSettings);

if(task.Status == Task.TaskStatus.Completed) {
  publishProgress("Downloading..");
  FileOutputStream fos = activity.
      openFileOutput(outputFile, Context.MODE_PRIVATE);
  try { restClient.downloadResult(task, fos); }
  finally { fos.close(); }
  publishProgress("Ready");
}
```

In subsection 2.4.2, it was explained how Abbyy requires a registration, in order to assign a user access to the limited space in the cloud. Every time the number of characters, that go through Abbyy's OCR engine, reaches the authorized limit, the free access is disabled. Then, a new e-mail account has to be used to get a new free license. For that reason, the application supports an easy way to enter and submit the new user's data. The timer that surrounds this asynchronous task intends to avoid receiving data that is taking too long to process. Normally, if the procedure takes too much time to finish, that should mean the result will not be correct and could lead to wasting a number of characters that would not be well recognized. So, the timer prevents these situations. The user only has to take another picture, paying more attention to the interested area.

When the message is ready, with no errors, the recognized characters are appended to a string buffer. After this operation, the text may proceed to be translated.

Abbyy makes it very simple to recognize characters. Its implementation and community help is very reasonable, although it still lacks in access to the cloud. However, when the result arrives, it is usually accurate.

4.3.2 Frame Translation: Vuforia

Getting a translation in realtime can aid the application's use and ease the role of the user. Hence the ability to do OCR instantaneously should give another perspective of Augmented Reality, one with greater simplicity, speed and user friendly. Thereafter, Vuforia is a software platform that operates with AR and gives the experience of seeing through a mobile App, allowing computer vision-based image recognition with capabilities that enhance the real world.

Vuforia offers several tutorials to help the developers, in [Lib15a]. Some preparations had to be made to run the Vuforia code. The Java API had to be built in Eclipse. Following the instructions to install and upgrade the software, available at [Lib15b; Lib15c], the *Vuforia Samples* project was imported into the workspace and the necessary files were added to the main project.

This project uses Vuforia 4.2.3 SDK. It offers *TextRecognition*, a sample code that returns the OCR of the existing characters within the camera sight. This sample starts to define a Region of Interest. As soon as the camera starts its preview, the captured frames are efficiently sent to the tracker. This tracker contains "sophisticated algorithms" ([Fan12]) to detect and track the image's natural features, in this case, the set of letters, and compare them with a resource database. The program is constantly updating the list of recognized words. Whenever a new word changes from the previous one, the translation method takes action and prints the result.

QCAR's method identifies each word as an object to be tracked independently. This means that multiple objects at the same Region of Interest can take its own time to be recognized. Therefore, the order of the words in the sentence can be random. It is up to the user to enter the words in the ROI according to their correct position, so he can get better semantic.

In this program, the ROI is a still area. However, to better fit the user's needs, its dimension should be dynamic. For example, if the region has a small height, that means it is expecting to read lines of words, whereas a larger height would be appropriate with paragraphs, as it is visible in figure 4.4, from [Lib15e]. A proper ROI that suits the requisites, improves the performance and pleases the user.

Figure 4.4: Vuforia's Region of Interest.

Also, the Vuforia's API available for implementation offers a default list file, with more than 100,000 English words, that should satisfy a basic search. In other words, this

App will detect and recognize English words. But the auto-detection feature of translation will allow to notice similar terms in other languages, although a better accuracy is achieved in English. It is possible to edit and add other languages' lists, but this was not considered relevant for the purpose of the project.

The text can be recognized in several styles, as mentioned in [Lib15d]. To get better results, the environment light should be uniform and the contrast between the text and the background should be enhanced.

Vuforia offers a modern and innovative way to capture the text from the real environment. Providing the software as open source, gives developers the possibility to create new applications for Augmented Reality in realtime, and apply new uses in various fields, such as gaming, education or self-aid. It still has to be improved in terms of languages, since English is the only one available but, overall, it offers an excellent perspective of AR and its accomplishments.

4.3.3 Translation: Microsoft Translator

Translating different languages is a bridge of communication between people from distinct countries and cultures. This link allows sharing information and knowledge to unify nations and connect the world. Although MT is not 100% accurate, it is still perceptive enough for the user.

As mentioned before, in subsection 2.5.2, both Google and Microsoft have developed their own translation APIs that can be very simply integrated inside developers' application codes. What distinguishes them, is mainly the price. According to Google's product manager Adam Feldman [Fri11], Google Translate API became deprecated on May 26, 2011 because "Translate API was subject to extensive abuse - the vast majority of usage was in clear violation of our terms. The painful part of turning off this API is that we recognize it affects some legitimate usage as well, and we're sorry about that". Therefore, as Microsoft Translate API is still offered as a free service, it was clearly selected to perform the translation matter of the project, as mentioned before in subsection 2.5.2.

Microsoft's APIs require a registration via email in Microsoft Azure Marketplace [Mica], in order to have access to the cloud. Once inside, an account key and a customer ID is given to the programmer. When introduced in the code, as shown in listing 4.4, he can choose any of the available APIs. Among the many existent resources is Microsoft Translator. After signing in, the user needs to register the application with a name and the Azure Marketplace credentials assigned. This process is explained in the tutorial [Micc] with more detail.

Listing 4.4: Microsoft Translator API Credentials.

```
MicrosoftTranslatorAPI.setClientId("MicrosoftID");
MicrosoftTranslatorAPI.setClientSecret("MicrosoftPW");
```

All the procedure can be slightly demanding, but the result is worthy. Besides, once again, the free cloud access has to be controlled to establish some limits for each user operational storage.

One of the advantages for using this kind of engine to translate, is the amount of supported languages, since there are already many available dialects for source and target requests, which is very convenient for a more universal application. Furthermore, as it comes from a world recognized company, the resources are constantly updated and the developer's community is continuously under assistance.

5

Results

This chapter resumes and compares the achieved results with the different methods to perform OCR in a Proof of Concept application. The idea of Augmented Reality facing translation is discussed around its advantages and disadvantages. There are several characteristics to have in mind when analysing the execution and the potential of the work. The sections above will consider the presented solution and make a contrast with other applications already on the market.

5.1 ARTrS: The Outcome

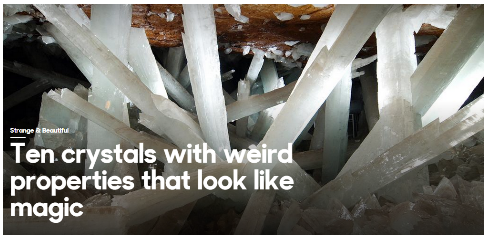
Several images were used for testing ARTrS application, and cover all the requirements to evaluate the best and worst points of Optical Character Recognition. Two types of measurements were studied: time to process and accuracy. The following items were considered:

- Image obstruction and noise;
- Styles and sizes of letters;
- Contrast between colours;
- Number of recognized characters;
- Speed of recognition;
- Words and sentences;
- Quality of translation.

The engines were tested with 20 pictures, in figure 5.1 above. They are composed by distinct features that cause various impacts on OCR. The purpose of these sample images is to study the performance and explore the advantages and disadvantages of the engines in a considerable amount of different pictures, as stated before.

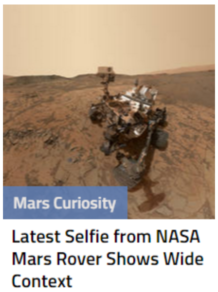
Sample A

Sample B



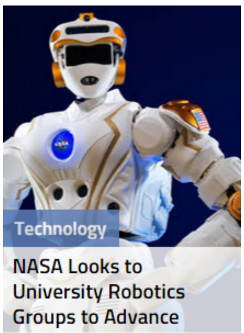
Sample C

Sample D



Sample E

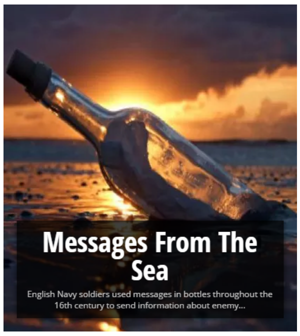
Sample F



Sample G

Sample H

Sample I



Sample J

Sample K

Sample L

Sample M

Sample N

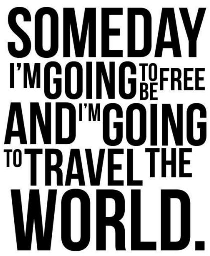
Sample O

Sample P

Sample Q

Sample R

Sample S



Sample T

Figure 5.1: Sample images proposed for OCR experiment.

5.1.1 OCR with Tesseract

The application contains an option to translate text by taking a picture and performing OCR with the Tesseract engine, followed by translation. The user should select the right source and target languages and focus the desired text in the middle of the camera, from the equipment in use. Then, take a photo of a selected area. The ROI is defined by pressing the finger at the beginning of the text, and dragging it until the end of the last character, creating a rectangle around the set of words. After a few seconds, the recognized text and translation are returned, as displayed in figure 5.2.

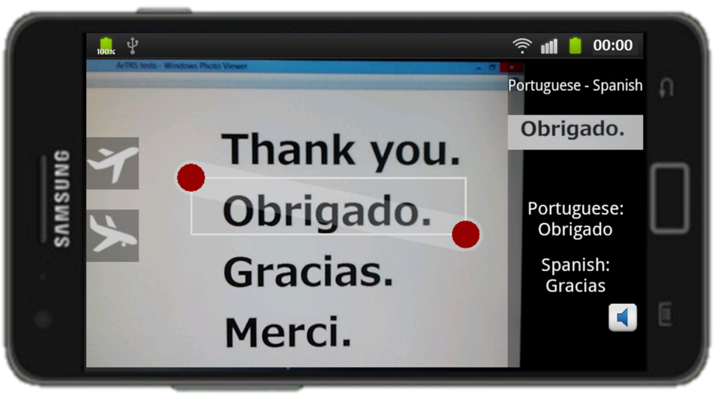


Figure 5.2: OCR and Translation performance with Tesseract.

Tesseract's first trait that is noticed is the contrast between the text and the background. The higher is the contrast, the more effective recognition is returned. The background should also be one-colour based, free of patterns. Black and white images are the best case scenario.

- Sample B shows an example of white letters on dark background. However, there are slight shades of bright light occasionally behind the text, which corrupts the recognition.
- Sample G, even with little background noise, represents a good scenario of a black and white image recognition.
- Sample Q, despite having a considerably good contrast, the disposition of the tiles can be noisy and weaken the recognition.

The next review characteristic was the letter's size and style. When there are words with different sizes in the same picture, if the difference is significant, only the text with best definition is well recognized. The other size characters can also be detected, although they are returned as garbage[1]. Different text styles do not have to be an issue either, since they keep on being similar to printed characters.

- Samples C and J represent a situation where the letters have very different sizes and only the best quality text is recognized. The first only recognizes larger letters, whereas the second detects smaller ones.
- Sample I shows that small differences of size should not be a problem.
- Sample L, for instance, reads the letters in the word "Nevada" as characters of different words, because of the surrounding frame style.

Symbols placed near the text can deceive the result, because they can be misunderstood as characters. If there is a frame captured along side the desired text, it can be associated to the words and damage the recognition.

- Samples M and O have arrows next to the text, which may deceive the engine.
- Samples O, Q and R are an example where the text is inside a frame that can harm the recognition.

Analysing simply words instead of full sentences is recommended for better results. Even splitting larger phrases into smaller ones can improve the performance. When recognizing a set of words, the first letter of each word is returned as a capital letter.

Since the orientation is handled, the text does not need to be exactly horizontally directed, but it should be close. The engine reads from left to right, from top to bottom.

[1] In Tesseract, garbage is usually a set of random characters, with no meaning.

This implementation of Tesseract does not read special characters, such as Arabic, Chinese, Korean, or Japanese characters.

The view should be focused, without any background noise, to get a good OCR. Even with the best capture properties, some letters can be misunderstood, like f-t and hi-m, among others. If there is no text in the area, the result is garbage.

Waiting time is not very long. The average is around 9 seconds, when a good quality picture is taken. It can diverge from good small images (around 4 seconds) to noisy pictures (20-25 seconds).

Overall, the best recognition requires mainly a good background, with a good contrast and focused letters. The larger is the ROI, the lower probability of accuracy. Unbound words are easier and more certain to perform a good OCR than sentences.

5.1.2 OCR with Abbyy

Another possibility for obtaining the translated text within a picture is by Abbyy's system. The process to get the translation of the text in hands is similar to the previous Tesseract method. Again, the text should be located at the centre of the screen to better fit its dimension. The user draws a proper rectangle around the area of interest and waits for the results. According to the source and target languages assigned, the translation is returned after some seconds, as figure 5.3 shows.

Figure 5.3: OCR and Translation performance with Abbyy.

Abbyy can usually handle background contrast very well. Only some cases where it is filled with drawings and different colour patterns, make the performance decrease. The captured picture should also avoid light reflexes, or the OCR will return nothing. Black and white images perform good OCR.

- Samples A, B and D are good examples of a good background, with acceptable results.
- Sample C, in turn, has a crowded background, which damages the recognition.

- Samples E, F and G have black text on white background, which reflects a good OCR.

It depends on the types of text, referred in [Typ15]. Otherwise, it does not perform OCR. The size of the letters should not be an obstacle to the recognition, unless the difference is big enough that the engine only detects the focused clean characters.

- Sample L has the word "NEVADA" in an unusual style, not recognised by Abbyy.
- Sample H displays a successful example of different sized letters, that do not disturb the recognition.
- Sample M only returns the lower sized capital letters.

Abbyy's engine appears to be more or less binary in terms of accuracy. In other words, it can return a good OCR, even with some errors, or nothing at all. These errors come from misunderstanding similar characters, such as t-!. Furthermore, long sentences can also cause mistakes in the recognition.

Lines and frames around the text may create some confusion to the engine, and the result is a blank string. Therefore, the user should be careful not to include them in the picture. On the other hand, symbols and unknown characters do not harm the recognition process. Abbyy seems to handle some possible rotation of the words.

When the text in the image is well captured, Abbyy performs a relatively fast and accurate OCR. However, its cloud access limits highly confines the user's usage of the software.

5.1.3 OCR with Vuforia

The third alternative to get the recognition and translation of printed text is by running Vuforia's software. This method performs OCR in realtime. So, the user only has to place the text inside the Region of Interest and wait. Once the text is tracked, the translation is received on the chosen target language, as disposed in figure 5.4.

Figure 5.4: OCR and Translation performance with Vuforia.

This method of Optical Character Recognition is usually very effective. The fact that it detects complete known words, means the accuracy has higher probability of being precise. The tracking feature is a way not to being continually translating the same word in the frame. That is to say, only when the set of tracked words that are returned are different from the previous ones, then the translating engine starts.

An aspect that seems to damage the recognition is the focus. If the camera does not get well focused in the Region of Interest, the probability of getting the correct recognition decreases. The difference of size from the characters should not cause any trouble either, although when some letters are too small or thin, they become blurred, and only larger letters can be detected (samples H and G are an example). If the calibration is weak, it misunderstands some characters like T-I, t-! and B-H.

Vuforia works well in various contrast situations. The background does not look to be an issue with low contrast scenes. However, settings with many objects behind the words may give detection some trouble.

- Sample B and C have several lightning and crystals behind the text, which decreases the process performance.
- Sample D best detected word was, precisely, "MOON", the one with white letters on light background.

Some other features were experienced:

- Samples A and D, for example, have special characters (":" and "?") that could not be recognized.
- Sample J, for instance, can clearly observe that short words, with less letters, are faster recognized.
- Sample L presents three styles of letters. The word "to" has a more discreet style, so it is more difficult to detect. The other styles were not hard to track.
- Samples N and S show that the software can handle the orientation very well.

In general, this is a very good engine. It takes around 1 and 2 seconds to do OCR after the text is tracked. Black and white images are well handled. Short words are faster to read. Sometimes, the best words to be recognized are the ones with lower contrast with the background. It can be very troublesome dealing with the right distance to the text, to get the best focus. The smaller and thinner the letters, the worst detection and recognition. Larger sized letters get better accuracy, only focus can damage the view. It can misunderstand the words' spacing and gather them in different order.

Vuforia requires the user to detect small sets of words, so that they can fit the Region of Interest. This aspect may affect the real sense of the text, since the translation will be directed to single words and not the full sentence.

5.1.4 Methods' Comparison

The real world presents adversities when performing both OCR and translation. The former may face obstruction of the view, stylised text, light contrast between characters and background and the possibility of naturally tremble the device while holding it. The latter can not run without internet connection, which may not be very convenient. A proper application should handle these issues and automatically adapt to the surrounding environment.

As mentioned before, several images were tested in order to find out the type o features supported by each of the Optical Character Recognition engines. Some of these features are described and evaluated in table 5.1.

It is important to refer that the table is just a resource to better distinguish Tesseract, Abbyy and Vuforia with the 20 case study images. The environment conditions may influence tests from different softwares to the same image. The selected features regard some of the best practices to have a good recognition.

The symbols used to characterize the features associated to each one of the engines just mean to distinguish them. That is to say, they do not imply that this feature is totally good or bad, but better or worse to each one of the engines.

Table 5.1: Comparison of different features between the OCR methods in study.

	Tesseract	ABBYY	Qualcomm vuforia
Number of available languages	✓	✓	✗
Handle slight contrast with the background	O	O	✓
Quantity of errors	✗	O	✓
Need good adaptable ROI	✓	✓	✓
Affected by symbols close to the text	O	O	✓
Handle focus	O	✗	✗
Handle thin/small letters	O	✗	✗
Handle brightness and rotation	O	O	✓
Handles long length text	O	O	O
Realtime	✗	✗	✓

The number of available languages is much more appealing to both Tesseract and Abbyy, than Vuforia itself. However, Vuforia is the only engine to process in realtime. The three engines need a flexible and adaptable Region of Interest for a better performance, but none is free from having issues to handle long and complex text. In general, Tesseract and Abbyy show similar acceptable results, whereas Vuforia got the best marks.

This analysis enhances some of the evaluated features as well as a comparison between Tesseract, Abbyy and Vuforia engines applied to different images. The idea was to collect as much diverse data as possible.

The following figures (5.5, 5.6 and 5.7) illustrate the performance of the studied engines in a graphical view. For each sample image, the time of process and the number of recognized characters were recorded. In some cases, if the text was too long, it would take more than one picture to catch the whole writing. This method might have captured the text under different conditions.

Tesseract should be preprocessed in terms of brightness and contrast, in order to better run the OCR engine. Frames and symbols near the text tend to deceive the output. Even under acceptable image conditions, Tesseract can misunderstand characters and make mistakes often. A good ROI improves the result. The number of available languages is also an advantage.

Below, figure 5.5 shows that Tesseract's levels of accuracy are not very consistent, mainly because of the image distortion. The more complex is the background, more errors are made, the longer it takes to process. Black and white text is favoured, as well as smaller length phrases.

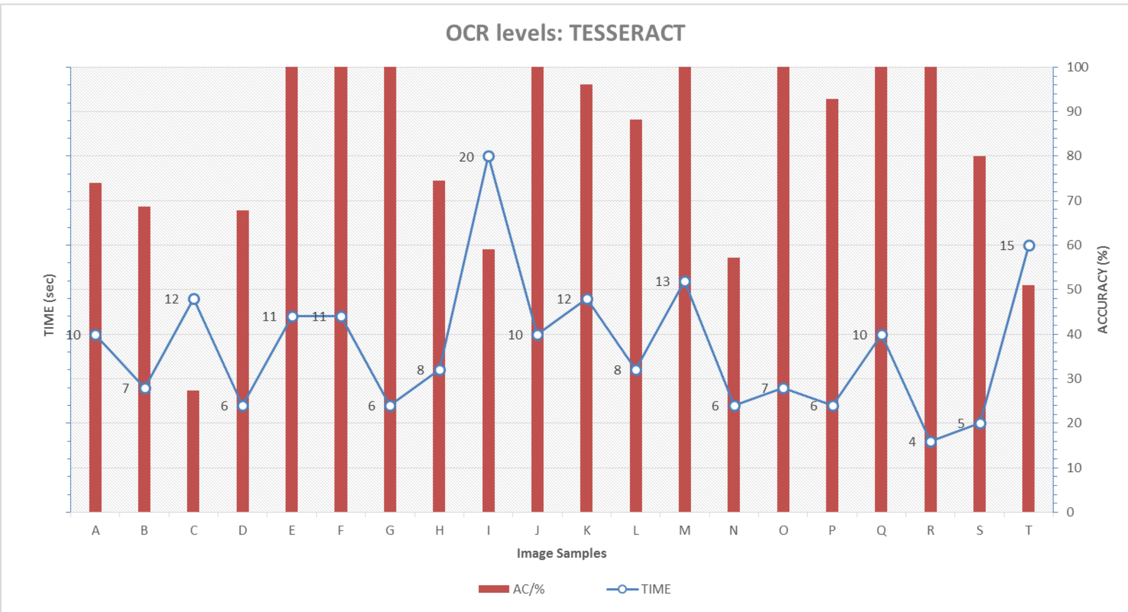


Figure 5.5: Tesseract testing accuracy and speed performance, for 20 sample images.

Abbyy returns very good results when the image is well prepared and focused. It does not make mistakes very often. Only if the text is long and the letters are small, there can be some misunderstanding now and then. Abbyy has a limit of characters to perform OCR, which can be very inconvenient. It takes less time to process simple and shorter words in a good background and the results are more correct. In general, this engine takes longer to return the final output.

The graphic in figure 5.6 reveals how the speed rates are higher, when the accuracy is not very good. It takes more time to run the engine if the picture is less clear. A more complex background may return nothing, because the system can not distinguish between the text and the scene.

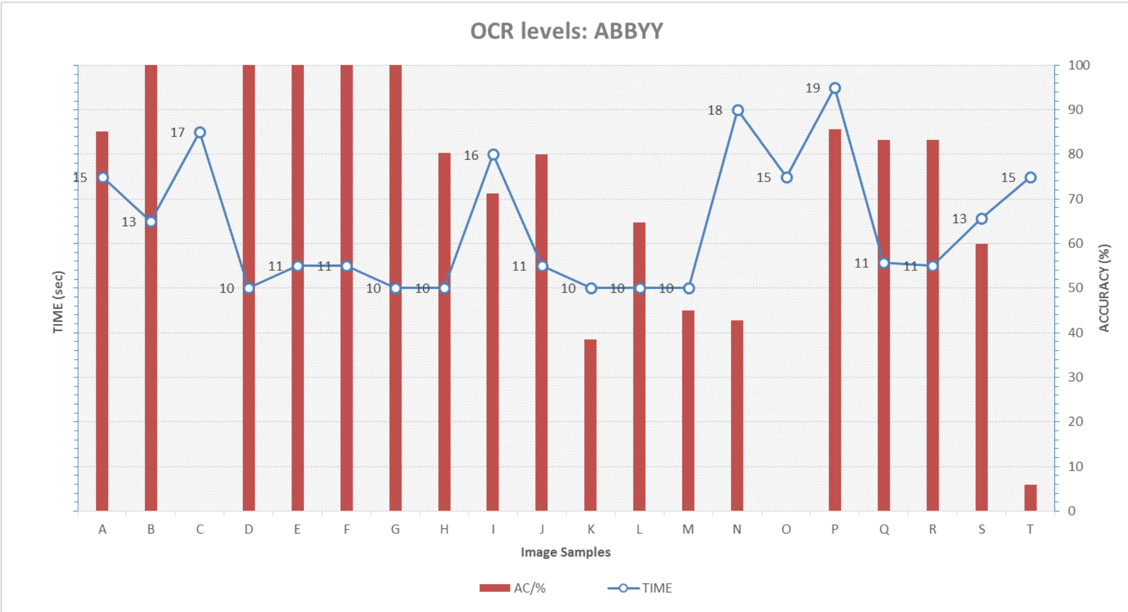


Figure 5.6: Abbyy testing accuracy and speed performance, for 20 sample images.

Vuforia has the advantage of offering realtime OCR, which is more user friendly. When running on smartglasses, it completely frees the hands of the user. The engine only provides English as a source language to recognize the text, which is not very general. When facing the real world, it handles natural distortions, such as brightness and rotation, very well. The camera should be well calibrated and focused, in order to allow Vuforia to accomplish a better recognition. The processing of thin sized letters may also decrease the performance.

Testing the sample images with Vuforia required the text to be split into sets of two or three words, so that they could better fit inside the ROI. Figure 5.7 verifies how the system takes very little time to recognize the characters, since the moment the words are being tracked. In terms of accuracy, it reveals that the algorithm is flexible for different situations, and around 75% of the cases were successful.

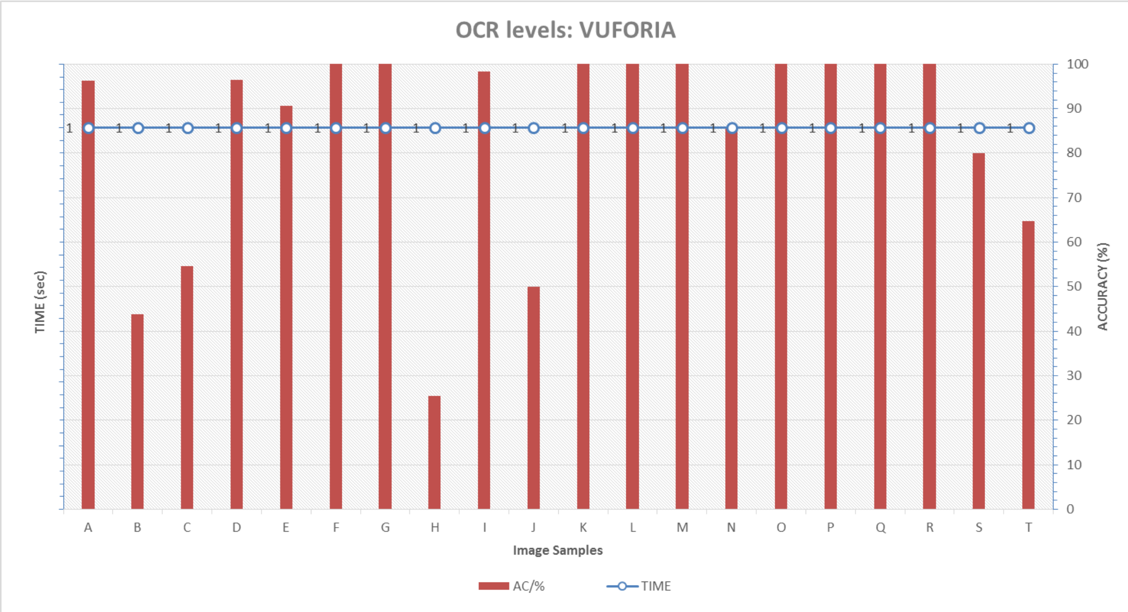
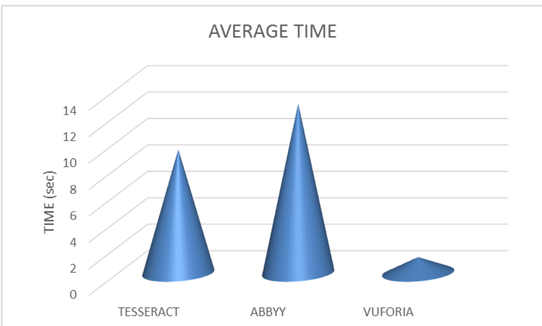
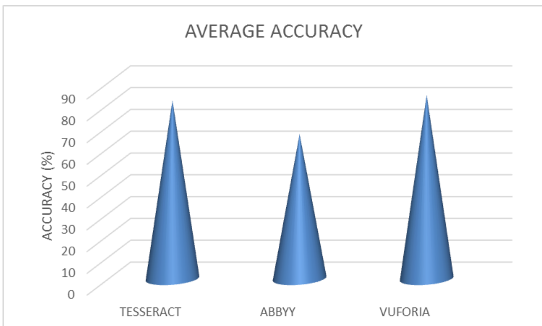


Figure 5.7: Vuforia testing accuracy and speed performance, for 20 sample images.

Finally, the average levels of time and accuracy to process are displayed in figures 5.8(a) and 5.8(b) respectively. Tesseract performs a balanced process between these features. Abbyy's engine is the one that gets the lowest accuracy and slowest processing time. Vuforia shows the best results, with a high rate of speed versus accuracy for a big range of samples.



(a) Time average levels.



(b) Accuracy average levels.

Figure 5.8: Average levels of OCR processing.

Overall, the studied engines perform OCR with more than 60% of accuracy and an acceptable waiting time to process the data. The translation also offers results ready for daily use. ARTrS application is able to be easily and quickly managed anywhere, since the existent menus support different features to do OCR and translation. The user is given the opportunity of having the traditional writing system or experience Augmented Reality in this modern field of translation.

5.1.5 Translation with Microsoft Translator

Being a Machine Translator, Microsoft Translator produces acceptable results. In a matter of one second or so, the requested sentence is returned in another language. Hiring someone to do this job would be more precise and trustworthy, but at the same time, it could become very expensive, complicated with a large number of languages and take too long for bigger projects. Although MT may be less accurate, the advantages outcome the disadvantages.

Performing OCR from a picture, implemented with Tesseract and Abbyy engines in this project, return good translations. However, the sentences can have recognition mistakes. In that case, Microsft Translator will not acknowledge the word, and the translation will not run properly, returning the same unresolved characters. The loss of information in the text, like an apostrophe, may also mislead the translator. Another issue that can cause some confusion, is the splitting of sentences to better fit the camera view, which will have a less accurate result.

Doing OCR in realtime with Vuforia, followed by Microsoft's translation, obtained great results in terms of speed. Accuracy is the feature that concerns the output. If the user detects the words in the correct order of the sentence, the translation will be precise and successful. Otherwise, the words can be correctly translated, but the meaning of the phrase will be lost.

In conclusion, Microsoft Translator may not have consistent translations in long length texts. The accuracy evolved in the last 10 years, but the quality still has to be highly improved. Nevertheless, it offers a wide range of different languages to use and fast process time with a quick translation output.

5.1.6 Testing on Device: Epson Moverio BT-200

Moverio BT-200 are Epson's new smartglasses, a modern technology that makes every recent user excited to experience. The glasses offer an easy environment and friendly User Interface (UI). The 3D feature to see pictures or watch movies is very interesting and the quality is satisfactory. It is able to use wireless connection to the internet and to other devices.

At first, the glasses may be awkward to wear, since they are a little heavy. The proximity of the screen can make the sight tired and dizzy, when used for too long. The manufacturer advises not to use the product while moving around. Furthermore, the equipment requires too many devices and cables. In addition to the smartglasses and the touchable instrument, the user still needs a cable to connect the two gadgets, another one to charge the widget or connect it to the computer and an extra one to link the headphones. The touch feature is not very sensitive, which makes it hard to manage the first movements on the device: dragging the mouse, double-tap selection, keyboard typing. Speech recognition is not allowed, because there is no microphone component. BT-200 is not very stylish either, making the user a little odd to the surrounding people, who can not see what is being displayed on the screen.

ARTrS takes advantage of the touch controller to navigate inside the application. If another smartglasses were to be used, like GoogleGlasses, which are only regulated by small buttons on the glasses, the App would have to be adapted to receive speech commands and touch free tasks. For the purpose, BT-200 were reasonable and efficient.

However, the general idea is that all the applications can run on both smartglasses and mobile devices. For this reason and because BT-200 is not quite independent from touchable commands, it may not be that useful with daily tasks. Smartphones are able to perform the same tasks and they are more accessible. Regardless of these facts, it should be praised that this technology is being developed, improved and launched today. It is very modern and sophisticated. Smartglasses are still far from taking full advantage of their expected futuristic usage, but they already taste an interesting portion of Augmented Reality.

5.2 ARTrS VS Commercial Applications

In subsection 2.3.1, some applications already on the market were researched and compared when performing Optical Character Recognition and translation. Three Apps were tested: OCR Instantly, CamDictionary and WordLens. They intended to be a reference, since their purpose are similar to the PoC. Below each one of the three Apps is aligned with the ARTrS PoC, in order to balance their features and usability.

5.2.1 OCR Instantly

OCR Instantly performs offline OCR and online translation. The experimented version was the free option. The OCR may misunderstand some characters, when the Region of Interest is not very well placed, even though the quality should be acceptable for a good output. Moreover, the biggest issue is the UI. It requires too many steps from the user to achieve the result: take a photo or get it from the gallery, manually calibrate the image levels (exposure, noise reduction and inverse the colour, if needed), save the changes and ask for OCR. Besides that and the advertisements always present, the App is relatively fast and the accuracy is acceptable.

ARTrS offers a similar option when performing Picture Translation with Tesseract or Abbyy. However ARTrS has a friendly UI, where the user is only requested to choose the appropriate menu, the source and target translation languages and select the interested area. The user does not need to be evolved with image processing adjustments.

5.2.2 CamDictionary

As the previous App, CamDictionary does offline OCR and online translation. The UI is very simple. The camera preview has a small pointer at the centre of the screen. According to the user's needs, the text can be extracted from a picture or a realtime frame.

The latter only requires the characters to be placed at the centre of the screen, behind the indicator. With a tap or just maintaining the view static, the camera is automatically calibrated and focused. Then, the OCR and translation of the present word are returned. The result is fast and accurate. The App allows the user to get dictionary lists, which can be free or paid versions.

ARTrS lacks on both accuracy and speed when compared with CamDictionary. However, it offers many more languages to recognize characters and translate.

5.2.3 WordLens

WordLens works offline for both OCR and translation features, which can be very convenient. It offers a considerably good amount of used languages. The procedure is very fast and accurate. An interesting characteristic that distinguishes this App from the others, is the possibility of seeing all the detected text in the screen, switched by its translation. In other words, the translated text layout tries to have a similar style, background and size to the original characters. Then, it is placed over the native words. However, the process can be too sensible to the user's natural hand shiver. With slight movements, the output keeps on appearing and disappearing, sometimes with a different recognition, and not giving the user enough time to read it. In general, WordLens application can give a very good experience of Augmented Reality. Once applied in the smartglasses, it may motivate AR use in daily life and occasional tasks.

It is difficult to compete with WordLens, since this App seems to be the most recent developed Augmented Reality Translation application, sponsored and applied by Google on Google Translate. The realtime engine is very efficient and the ability to camouflage the original phrase with the new translated characters really outlines AR to its best. However, ARTrS realtime menu, with Vuforia, freezes the recognized text enough time to allow the user to read the translation, which surpasses WordLens' lack of processing pause and tracking.

6

Conclusion and Future Work

6.1 Lessons Learned

The purpose of this dissertation was to converge the Translation perception with a modern concept, Augmented Reality. The main reason was to answer a common issue of communication in our multicultural world, more and more connected and engaged everyday, every way. It also meant to help the RICS group to acquire knowledge in the techniques and development environments used in AR.

An idea to achieve this goal was to create an application, that could offer a potential customer the possibility to travel abroad with an instant translation tool. This mechanism should allow the user to look around and read all the surrounding information in his own language, through a device. For better accessibility, it should run on smartphones. On the other hand, in order to fully experience Augmented Reality, it would run on a new technology: smartglasses.

It is important to outline that one year ago, by September 2014, this target was still a notion to be studied and investigated. One of the first concerns was the need to explore a new concept, which was yet to raise and bloom. However, many developers have recently been working and researching this subject. Since then, both companies and independent programmers have launched new applications. Developers have been testing and improving the possibilities of this area.

Although Augmented Reality has been around for some time, it has just been recently releasing good results on the market. New software and hardware have been tested, and final products have been getting the internet closer to our lives. Technology keeps on being more incorporated in our daily routines. Fields like Medicine, Education, Army, Production, Manufacturing, Marketing, along with many others, have been highly studied and improved.

The software perspective was to detect the text within the screen view, capture the letters by doing Optical Character Recognition, and return the respective translation. The expected results envisioned good accuracy, fast processing rate and image versatility. It was tested in the real world and with 20 different pictures, that could cover several features to evaluate image processing.

Optical Character Recognition is the process to detect and get the text out of an image. Between the available techniques, this system was implemented for captured pictures and realtime video framing. The former was accomplished with Tesseract, a free software from Google that works offline, and Abbyy, an API with cloud limited access. The latter is driven by Vuforia text recognition tool. The mechanism to take a picture and process it takes longer and is more daring to commit errors, whereas computer vision-based recognizes word by word and releases the user from tapping tasks, enhancing the real world view. The three engines are more efficient when the Region of Interest is reduced to fit the text inside. Furthermore, they all have an active community to assist developers programming and enrolling new software.

As expected, Machine Translation revealed to be a very useful tool when translating words in signs and titles. Long text may decrease the probability to keep the sense of the phrase. Between Google and Microsoft very similar translator engines, the latter offered a more accessible API in terms of budget. For this reason, as a free service, Microsoft Translate API was chosen to be implemented in the Proof of Concept. Despite the registration requirement and the requesting limitations, the results were very acceptable in processing time and quality. Moreover, the engine has various languages available, as well as an auto detection feature that automatizes the general process.

Wearing the smartglasses brought a new perspective to Augmented Reality. Despite the unusual style and the uncomfortable weight on our face, Moverio BT-200, from Epson, can be quite impressive. The ability to have our hands completely free for other tasks, while wearing the glasses, is very useful. The user is offered an experience to see the world complemented with digital knowledge. New solutions bring stylish and light glasses, as well as more precision and stability to focus the image view.

The dissertation is complemented with Annex A: User Manual, intended to help the user managing the Proof of Concept application. Each menu and its characteristics are described, as well as the available steps and actions.

To sum up, being able to perform translation through Augmented Reality is a technology yet to be evolved and matured, since it has so much to offer. The following points outline the behaviour of the technology, as well as the main essence of the leading steps of the project:

- OCR requires a lot of image processing that decreases the speed performance. It needs to become more adaptable to natural adversities, such as hidden letters or brightness issues. The accuracy and speed of the response may be acceptable for now, but it still has a long way to be improved, towards an instantaneous and certain data output.

- The translation system works very well when considering single words. However, it lacks on keeping the sense of a full sentence.
- The developers are highly supported with software and forums to inspire new ideas.
- AR appears to be a wide field, full of resourceful tools, possibilities and applications in our daily lives. It gives the user the ability to be closer to the information and presence the world in a sophisticated way.

Overall, having an application available to transform the foreign surroundings into our own known world will break walls of difference. The utility of these applications envisions a scenery where the Internet of Things (IoT) is a reality. Communication and comprehension can get universal. People and their cultures may get closer. Nowadays, technology is always changing and evolving. New inventions keep coming out very fast, with more quality and creativity. The next step on this matter is to translate speech in realtime. "Talking with people all over the world is now possible with Skype Translator". This is the most recent outcome for video and voice realtime Translation.

6.2 Next Steps

Follows some of the weaknesses to be improved and upgraded:

- Text-to-speech component does not support several languages available for translation.
- Vuforia only detects and recognizes English language.
- Tesseract is supported by more languages than the ones offered by the Proof of Concept application.
- Tesseract does not read special characters, such as Japanese.
- Moverio smartglasses do not support voice commands, so the second menu, Speech, is useless with this device.
- The sample images might have been treated under different conditions, which could have led to less constant output.
- The software should become more flexible and adaptable to the surrounding environment.

A few features were also meant for future implementation. The primary aspects would be the ones below:

- Simple activities of help to guide the user through the application.
- Database to save read and translated words.
- Allow the device to have hand recognition, in order to select the Region of Interest before taking a picture, or further commands.
- Create menus: Picture-to-Word and Word-to-Picture. The first enters a picture into the database, recognizes the object by image search and returns the translated name. The second does the opposite, by entering a word and returning the first three images, for instance, of the object.

Many other features may be improved and implemented, but the ones referred previously are a first step to endorse the work with quality and creativity. Augmented Reality has a wide field to explore and developers are not afraid to get out of the box to try new areas. Technology is progressing and expanding the limits of Reality.

Bibliography

[Abba] Abbyy. *Abbyy Cloud OCR SDK*. URL: http://ocrsdk.com/.

[Abbb] Abbyy. *ABBYY Cloud OCR Service Architecture*. URL: https://abbyy.technology/en:products:cloud-ocr:cloud-service-architecture.

[Aic13] A. Aich. *Augmented Reality: Vision Redefined*. 2013. URL: http://www.maacindia.com/blog/index.php/augmented-reality-vision-redefined/.

[All15] A. Allsopp. "Google Glass 2 UK release date, price and specification rumours: New snap-on design for second-gen Google Glass smartglasses". In: *TechAdvisor* (2015). URL: http://www.pcadvisor.co.uk/news/wearable-tech/google-glass-2-release-date-price-specs-not-dead-io15-hires-design-3589338/.

[Alv15] J. Alvarez. "HoloLens price could be out of reach for some people, says Microsoft executive". In: *Digital Trends* (2015). URL: http://www.digitaltrends.com/computing/microsoft-hololens-will-cost-significantly-more-than-a-vidoe-game-console/.

[Ope] *Android OpenCV Manager: Introduction*. 2015. URL: http://docs.opencv.org/2.4.11/platforms/android/service/doc/Intro.html#architecture-of-opencv-manager.

[Chi07] A. Chitu. *Google Switches to Its Own Translation System*. 2007. URL: http://googlesystem.blogspot.pt/2007/10/google-translate-switches-to-googles.html.

[CM13] B. B. Christopher Mohr. *Qualcomm to Add Text Recognition Technology to Vuforia Platform*. 2013. URL: http://education.tmcnet.com/topics/education/articles/322187-qualcomm-add-text-recognition-technology-vuforia-platform.htm.

[Con15] A. C. O. S. Console. *Abbyy*. 2015. URL: https://cloud.ocrsdk.com/Account/Register.

[Cra10] C. Craft. *Augmented Reality - A Simple Explanation*. 2010. URL: https://www.youtube.com/watch?v=KFLb8BZQ6_I.

[DP15] A. media Developer Portal. *What is the ARmedia 3D SDK main purpose?* 2015. URL: http://dev.inglobetechnologies.com/support/choose.php.

[Dev15] G. Developers. *Google Maps SDK for iOS - Markers*. 2015. URL: https://developers.google.com/maps/documentation/ios/marker.

[DA15] S. G. Dhiraj Amin. "Comparative Study of Augmented Reality SDK's". In: *International Journal on Computational Sciences & Applications (IJCSA)* 5.1 (2015).

[Doc15] C. O. S. Documentation. *Abbyy Cloud OCR SDK*. 2015. URL: http://ocrsdk.com/documentation/.

[Due14] B. L. Due. "The future of smart glasses: An essay about challenges and possibilities with smart glasses". In: *Centre of Interaction Research and Communication Design, University of Copenhagen* (2014). URL: http://circd.ku.dk/images/An_essay_about_the_future_of_smart_glasses.pdf.

[Ega15] M. Egan. "What is Oculus Rift? Why Oculus Rift matters, when you can get Oculus Rift". In: *TechAdvisor* (2015). URL: http://www.pcadvisor.co.uk/buying-advice/gadget/3522990/oculus-rift-release-date-specs-features/.

[Fan12] Y. Fan. "Mobile Room Schedule Viewer using Augmented Reality". In: *Study Programme for a Degree of Bachelor of Science in Computer Science* (2012). URL: http://hig.diva-portal.org/smash/get/diva2:535103/FULLTEXT01.pdf.

[Fre11] K. Freenman. *QuickAdvice: CamDictionary Translates 54 Languages - Plus, Win A Copy!* 2011. URL: http://appadvice.com/appnn/2011/07/quickadvice-camdictionary-translates-54-languages-plus-win-a-copy.

[Fri11] I. Fried. "Google not killing translate API, will develop paid version". In: *CNET* (2011). URL: http://www.cnet.com/news/google-not-killing-translate-api-will-develop-paid-version/.

[FN14] M. Y. Fritz Nelson. "The Past, Present, And Future Of VR And AR: The Pioneers Speak". In: *Tom's Hardware, The Authority on Tech* (2014). URL: http://www.tomshardware.com/reviews/ar-vr-technology-discussion,3811-3.html.

[Goo15a] S. Goodman. *Machine Translation – Google Translate vs Bing Translator*. 2015. URL: http://blog.linnworks.com/google-bing-translate/.

[Goo15b] Google. *tesseract-ocr*. Google Code. 2015. URL: https://code.google.com/p/tesseract-ocr/.

[Gra14] A. Grace. *Optical Character Recognition... quick and helpful*. 2014. URL: http://pt.appszoom.com/android_applications/tools/ocr-instantly-pro_ikijx.html.

[Gro] G. Groups. *Teseract vs Abbyy*. URL: https://groups.google.com/forum/#!topic/tesseract-ocr/i_102U2GONg.

[Tesa] *How To Make Simple OCR Android App Using Tesseract*. 2012. URL: http://kieplaptrinhvien.blogspot.pt/2014/12/how-to-make-simple-ocr-android-app.html.

[Hut05] J. Hutchins. "The history of machine translation in a nutshell". In: (2005).

[Int15] IntSig. *CamDictionary*. 2015. URL: https://play.google.com/store/apps/details?id=com.intsig.camdict&hl=en.

[Iov15] A. Iovkova. *McDonald's Relies on ABBYY OCR Technology to Power Mobile App*. 2015. URL: http://blog.ocrsdk.com/mcdonalds-relies-on-abbyy-ocr-technology-to-power-mobile-app/.

[joo08] joos2322. *Wikitude AR Travel Guide (Part 1)*. 2008. URL: https://www.youtube.com/watch?v=8EA8xlicmT8.

[LK12] J. Lambrecht and J. Kruger. "Spatial Programming for Industrial Robots based on Gestures and Augmented Reality". In: *IEEE/RSJ International Conference on Intelligent Robots and Systems* (2012).

[Lib15a] Q. V. D. Library. *Getting Started*. 2015. URL: https://developer.vuforia.com/library/getting-started.

[Lib15b] Q. V. D. Library. *How To Compile and Run a Vuforia Android Sample*. 2015. URL: https://developer.vuforia.com/library/articles/Solution/Compiling-and-Running-a-Vuforia-Android-Sample-App.

[Lib15c] Q. V. D. Library. *Installing the Vuforia Android SDK*. 2015. URL: https://developer.vuforia.com/library/articles/Solution/Installing-the-Vuforia-Android-SDK.

[Lib15d] Q. V. D. Library. *Text Recognition*. 2015. URL: https://developer.vuforia.com/library/articles/Training/Text-Recognition-Guide.

[Lib15e] Q. V. D. Library. *Text Recognition Tips*. 2015. URL: http://developer.vuforia.com/library/articles/Best_Practices/Text-Recognition-Tips.

[LLC14a] M. L. LLC. *Brain Scan Augmented Reality Education App – Case Study*. 2014. URL: http://www.marxentlabs.com/ar-videos/augmented-reality-education-app-case-study/.

[LLC14b] M. L. LLC. *Moosejaw X-Ray Augmented Reality Catalog – Case Study*. 2014. URL: http://www.marxentlabs.com/ar-videos/moosejaw-xray-app-augmented-reality-case-study/.

[Mar13] MarkBangs. *CamDictionary Free app review: translate between several of the most popular languages*. 2013. URL: http://www.apppicker.com/reviews/8406/CamDictionary-Free-app-review-translate-between-several-of-the-most-popular-languages.

[Mar15a] M. A. Marketplace. *Microsoft Translator*. 2015. URL: https://datamarket.azure.com/dataset/bing/microsofttranslator.

[Mar15b] C. Martin. "Google Glass UK release date, price and specs: Google Glass will go offsale on 19 January; how to buy Google Glass". In: *TechAdvisor* (2015). URL: http://www.pcadvisor.co.uk/feature/gadget/google-glass-release-date-uk-price-specs-3436249/.

[Mar15c] C. Martin. "Microsoft HoloLens UK release date, price, specifications and games rumours: HoloLens developer edition expected in 2016". In: *TechAdvisor* (2015). URL: http://www.pcadvisor.co.uk/new-product/wearable-tech/microsoft-hololens-release-date-price-specs-rumours-3616014/.

[McK15] V. McKalin. *Augmented Reality vs. Virtual Reality: What are the differences and similarities?* 2015. URL: http://www.techtimes.com/articles/5078/20140406/augmented-reality-vs-virtual-reality-what-are-the-differences-and-similarities.htm.

[Mica] Microsoft. *Microsoft Azure*. URL: https://datamarket.azure.com/.

[Micb] Microsoft. *Microsoft Azure Marketplace*. URL: https://datamarket.azure.com/home.

[Micc] Microsoft. *Walkthrough: Signing up for Microsoft Translator and getting your credentials*. URL: http://blogs.msdn.com/b/translation/p/gettingstarted1.aspx.

[Mic15] Microsoft. *Microsoft HoloLens - Transform your world with holograms*. 2015. URL: https://www.youtube.com/watch?v=aThCr0PsyuA.

[Mil14] C. H. Miller. "Epson releases second-gen Android-powered Moverio BT-200 smart glasses for $699.99". In: *9to5Google* (2014). URL: http://techwonda.com/epson-reality-glasses-moverio-bt-200-on-sale/.

[Mue14] F. Mueller. "Enhancing Eclipse for introductory programming". In: (2014). URL: ftp://ftp.cs.purdue.edu/pub/hosking/papers/mueller.pdf.

[Nay] M. Nayak. *How the Internet of Things Is Shaping Our Future*. URL: http://tech.co/internet-of-things-shaping-future-2014-11.

[New14] D. Newcomb. *Eyes On With Heads-Up Display Car Tech*. 2014. URL: http://www.pcmag.com/article2/0,2817,2461037,00.asp.

[Oli15] P. M. Oliveira. “Realidade Virtual”. In: *Exame Informática* 236 (2015), pp. 62–69.

[O.M13] H. O.Matei P.C.Pop. “Optical character recognition in real environments using neuralnetworks and k-nearest neighbor”. In: *Springer Science+Business Media New York 2013* (2013).

[Pla15a] G. C. Platform. *Pricing*. 2015. URL: `https://cloud.google.com/translate/v2/pricing`.

[Pla15b] G. C. Platform. *Translate API FAQ*. 2015. URL: `https://cloud.google.com/translate/v2/faq`.

[Pra15] L. Prasuethsut. “Epson Moverio BT-200 review: Augmented reality is still far away, and these smart glasses prove it”. In: *Techradar* (2015). URL: `http://www.techradar.com/reviews/gadgets/epson-moverio-bt-200-1212846/review`.

[Pri13] E. Price. *MARS App Uses Augmented Reality to Help Identify Things Around You*. 2013. URL: `http://mashable.com/2013/03/08/par-works-mars-app/`.

[Pro] ProgrammableWeb. *ABBYY Cloud OCR API*. URL: `http://www.programmableweb.com/api/abbyy-cloud-ocr`.

[Qua15] Qualcomm. *Dev Guide: Vuforia SDK Architecture*. 2015. URL: `https://ui-dev2.vuforia.com/resources/dev-guide/vuforia-ar-architecture`.

[Rea09a] V. Reality. *Future of augmented reality*. 2009. URL: `http://www.vrs.org.uk/augmented-reality/future.html`.

[Rea09b] V. Reality. *When was augmented reality invented?* 2009. URL: `http://www.vrs.org.uk/augmented-reality/invention.html`.

[Rid13] P. Ridden. *IKEA catalog uses augmented reality to give a virtual preview of furniture in a room*. 2013. URL: `http://www.gizmag.com/ikea-augmented-reality-catalog-app/28703/`.

[Sch07] B. Schwartz. *Google Translate Drops Systran For Home Brewed Translation*. 2007. URL: `http://searchengineland.com/google-translate-drops-systran-for-home-brewed-translation-12502`.

[Sch14] H. Schweizer. “Smart glasses:technology and applications”. In: (2014).

[sci] e sciencecity. *Why use clouds?* URL: `http://www.cloud-lounge.org/why-use-clouds.html`.

[Sdk13] A. C. O. Sdk. “ABBYY Cloud Solution Helps Aetopia Modernise Directories for the Public Records Office of Northern Ireland”. In: *Abbyy Case Study* (2013). URL: `http://www.abbyy.com/media/1186/cs_aetopia_proni_cloudocr_sdk_e.pdf`.

[Sdk15] A. C. O. Sdk. *OCR for Android, iPhone and any other mobile device*. 2015. URL: http://ocrsdk.com/producttour/mobile-devices/.

[SP15] S. C. Sean Peasgood. "Augmented Reality Glasses Getting Closer To Reality". In: *cantech letter* (2015). URL: http://www.cantechletter.com/2015/06/augmented-reality-glasses-getting-closer-to-reality/.

[Sef15] G. I. Seffers. "Smart Glasses To Augment Battlefield Reality For U.S. Marines". In: *AFCEA* (2015). URL: http://www.afcea.org/content/?q=Article-smart-glasses-augment-battlefield-reality-us-marines.

[Smi07a] R. Smith. "An Overview of the Tesseract OCR Engine". In: *ICDAR* (2007). IEEE Ninth International Conference.

[Smi07b] R. Smith. *Tesseract OCR Engine*. 2007. URL: https://tesseract-ocr.googlecode.com/files/TesseractOSCON.pdf.

[Soc15] SocialCompare. *Augmented Reality SDK*. 2015. URL: http://socialcompare.com/en/comparison/augmented-reality-sdks.

[Soc11] A. D. Society. *"App" voted 2010 word of the year by the American Dialect Society (Updated)*. 2011. URL: http://www.americandialect.org/app-voted-2010-word-of-the-year-by-the-american-dialect-society-updated.

[Son] "Sony steps forward with smart glasses". In: *The Japan Times News* (2015). URL: http://www.japantimes.co.jp/news/2015/02/18/business/corporate-business/sony-steps-forward-with-smart-glasses/#.VWwxxc-4Tct.

[Staa] Stacey. *Google or Bing Translator: Which One Should You Choose?* URL: http://www.omniglot.com/language/articles/onlinetranslators.htm.

[Stab] I. L. Stats. *Internet Users*. URL: http://www.internetlivestats.com/internet-users/.

[Sun11] D. Sung. "The history of augmented reality". In: *Pocket-lint* (2011). URL: http://www.pocket-lint.com/news/108888-the-history-of-augmented-reality.

[Sup15] G. P. H. Support. *ReadOnlyTransition*. 2015. URL: https://code.google.com/p/support/wiki/ReadOnlyTransition.

[Swi15] M. Swider. "Google Glass review". In: *Techradar* (2015). URL: http://www.techradar.com/reviews/gadgets/google-glass-1152283/review.

[Tay15] B. Taylor. "3D smart glasses will transform workflows around the world, says Atheer's CEO". In: *TechRepublic* (2015). URL: https://www.yahoo.com/tech/s/3d-smart-glasses-transform-workflows-222441134.html.

[Tesb] *tesseract-ocr*. URL: https://code.google.com/p/tesseract-ocr/downloads/list.

[The] R. Theis. *Fork of Tesseract Tools for Android*. URL: https://github.com/rmtheis/tess-two.

[The15] TheSimplest.Net. *OCR Instantly Free*. 2015. URL: https://play.google.com/store/apps/details?id=com.thesimplest.ocr&hl=en.

[Tra15] T. Tran. *AR with Vuforia*. 2015. URL: http://www.academia.edu/7028565/AR_with_Vuforia.

[Tur15] B. Turovsky. "Hallo, hola, olá to the new, more powerful Google Translate app". In: *Google Official Blog* (2015). URL: http://googleblog.blogspot.co.uk/2015/01/hallo-hola-ola-more-powerful-translate.html.

[Typ15] C. O. S. D. T. Types. *Abbyy Cloud OCR SDK*. 2015. URL: http://ocrsdk.com/documentation/specifications/text-types/.

[Ula15] L. Ulanoff. "Hands on with Google Translate: A mix of awesome and OK". In: *Mashable* (2015). URL: http://mashable.com/2015/01/14/hands-on-with-google-translate/.

[Ume14] S. Ume. "Epson Augmented Reality SmartGlasses, Moverio BT-200 on Sale For $700". In: *TechWonda* (2014). URL: http://techwonda.com/epson-reality-glasses-moverio-bt-200-on-sale/.

[VR15] O. VR. "First Look at the Rift, Shipping Q1 2016". In: (2015). URL: https://www.oculus.com/blog/first-look-at-the-rift-shipping-q1-2016/.

[Wik15a] Wikipedia. *Augmented reality*. 2015. URL: http://en.wikipedia.org/wiki/Augmented_reality#History.

[Wik15b] Wikipedia. *Google Translate*. 2015. URL: http://en.wikipedia.org/wiki/Google_Translate.

[Wik15c] Wikipedia. *History of machine translation*. 2015. URL: http://en.wikipedia.org/wiki/History_of_machine_translation.

[Wik15d] Wikipedia. *Microsoft Translator*. 2015. URL: http://en.wikipedia.org/wiki/Microsoft_Translator.

[Wik15e] Wikipedia. *Mobile app*. 2015. URL: http://en.wikipedia.org/wiki/Mobile_app.

[Wis11] H. Wise. "Word Lens Introduces French Language to Its Augmented Reality-Based Translation Capabilities". In: *Market Wired* (2011). URL: http://www.marketwired.com/press-release/word-lens-introduces-french-language-its-augmented-reality-based-translation-1598714.htm.

A

ARTrS: User Manual

ARTrS was the name chosen for the PoC application and it means Augmented Reality TranSlation. When the user opens the App, he is introduced to a GridView menu that presents all the submenu options, displayed in figure A.1. The two main purposes of this project are Optical Character Recognition and translation, through Augmented Reality. Each menu covers at least one of these goals.

Figure A.1: Main menu layout.

Menus 1 and 2 are both based on traditional translation resources. They do not include Augmented Reality features, only ordinary translation, already available on Google Translate or Microsoft Translator. The user should type a set of input words to be translated and returned as output. These menus shaped a first draft and a base to begin the App.

The two following menus, 3 and 4, are much more elaborated. They already include both objectives, OCR and translation with AR. Although very similar, they were implemented and run in different ways, which gives them both a path of contrast and comparison. Their general process includes taking a picture of the intended text, running it through an OCR engine and translating the detected words.

Finally, the last menu performs a realtime translation, where the user just points the camera at the desired text to get an instant translation. An in-depth manual of the process of each menu is presented below.

A.1 Menu 1: Text

The first menu is called Text. The process is relatively simple. It works as the traditional translator found online at Google or Microsoft engines and its goal is to create a first approach to a translator. It can also be seen as a bilingual dictionary search. In other words, according to the source and target languages and given the input words, some output will match.

In this case, the user needs to type the words meant to translate and choose a target language. Then, just by clicking on the Translate button, the respective output appears. The source language is detected automatically, but it can be changed if the detection does not seem to be fully accurate to the truth. Both source and target languages are initially set to English. There is still a sound button which, once pressed, reads the translation out loud, in order to give the user the ability to correctly say the words.

A.2 Menu 2: Speech

Next, is Speech, very similar to the previous menu. However, it receives the user's spoken words as input. The speech-to-text operation returns several options of recognition, where the one with higher probability of matching, is the one that is written as input. If the user clicks on the recognized text, a list view appears with the other probable speech options, organized in decreasing order of matching probability. This way, the best entering text can be chosen.

As before, the text can be translated according to the detected source language and the chosen target language. Clicking on the Translate button triggers the translation process and returns the translated text. A sound button is available as well to play the correct reading of the output.

A.3 Menu 3: Photo Tesseract

Photo: Tesseract is a menu where the user is able to face Augmented Reality. It requires the user to take a photo of the desired text to be processed. The main screen shows the camera view and the user just has to point the camera at the text and select it. To define the area, the finger must be pressed and dragged from the beginning until the end of the intended section, drawing a rectangle around the whole text. The purpose for limiting the view is to acquire speed and accuracy processing.

After releasing the touched area the processing is triggered. Here the source and target languages need to be defined from the beginning. A picture of the whole screen is taken and saved. The image is cut with the rectangle coordinates, drawn by the finger. The smaller image is converted into grey scale, to ease the next steps, and it is sent into the Tesseract's engine to get the text. Tesseract method is a very useful tool supported by Google, to do Optical Character Recognition. It works both online and offline.

After doing the text recognition, follows the translation method, already described. Once again, there is a text-to-speech button available.

A.4 Menu 4: Photo Abbyy

Photo: Abbyy works the same way as menu 3 with the exception that it uses the Abbyy engine instead of Tesseract's. As before, a picture of the full screen needs to be taken. After that, the user taps with the finger to delineate the ROI. This photo is cut, fitting the rectangle defined by the pressed coordinates, so that the processing can be easier and faster. For the same reason, the smaller image is converted into grey scale. Then, Abbyy engine takes place to do OCR and get the existing text. After that, the translation takes action. When the task is complete, the output text appears, followed by the text-to-speech button resource.

A.5 Menu 5: Realtime

The last menu is Realtime. This menu uses Vuforia sample code to do OCR in realtime, supported by Vuforia API. The screen shows the camera view. There is a ROI where the user should point the text at. If there are any characters in the area, a green rectangle appears around each recognized word. There are some text styles supported by Vuforia, referenced in [Lib15d]. This ROI can be pressed to force focusing the area. As soon as the text is returned from Vuforia's cloud, the translation starts. In order to avoid continuous translating the text, this request only takes place if the actual set of words are different from the ones detected previously.

Vuforia SDK only incorporates a list of 100,000 English words. This list can be improved by the developer, following some conventions detailed in [Lib15d]. However, for the purpose of this project, the extension of the dictionary was not a priority. That is why the source language in this menu is restricted to English. The target language keeps the large range of options, since it only depends on the Microsoft Translator resources. It can be chosen from a slide menu activated by a button placed on the bottom left corner of the screen.

The text is always being updated on the UI. Once again, there is a text-to-speech button available to correctly read the output.