

**Wydział Inżynierii Mechanicznej i Robotyki**

Katedra Robotyki i Mechatroniki

Praca Inżynierska

*Project of an „AGH Guide” application based on the augmented reality.*

*Projekt aplikacji „Przewodnik AGH” bazującej na rozszerzonej rzeczywistości.*

Autor: Filip Babiec

Kierunek studiów: Inżynieria Mechatroniczna (w języku angielskim)

Opiekun pracy: dr inż. Rafał Radecki

Kraków, 2022

Table of contents

[**Introduction** 3](#_Toc87876224)

[**1.** **Market analysis** 4](#_Toc87876225)

# Introduction

Advances in technology along with the dynamically developing market of mobile applications have resulted in an increased popularity in AR and VR technologies, i.e. technologies that connect the real world with the virtual world. They gather most popularity in the world of mobile and computer games industry, but they also find applications in other fields like simulators or military. More and more of these technologies help us not only in entertainment or specific environment but either in everyday life, for instance by navigating us around the world. In addition the development of more powerful hardware allowed to create even more immersive and diverse experiences with virtual and augmented realities.

Hence in this Engineering Thesis the creation of a location based AR application has been undertaken. The purpose of this application is to guide new members of AGH university and freshmen around the AGH campus. This app is developed for most used mobile operating system in the world – Android. It uses the end device GPS system and clock to determine the user’s position and time in the real world and then to navigate him through the AGH buildings. It also incorporates user’s main camera to project virtual objects into real world like arrows or markers to help navigating, or display basic information about AGH buildings around.

The first chapter contains the functionality of the application. In the second chapter the market analysis for existing solutions was made as well as the inspection of tools for application development. The third and fourth chapters consists of description of development process, and the last chapter presents the conclusions.

# 1. Scope of work

The aim of this work consists of two main tasks, first to create a navigation application that will use end device GPS data to determine user’s position, and the second one to implement the AR features to the navigation. Additionally the app will have the game mode where you complete different tasks by walking between locations. The main features of navigation are navigating to chosen building with specific floor determined by the room the user chose, navigation to each department’s dean’s office and displaying information about selected building. The AR functionality will cover navigation in the AR environment which will use end device camera, displaying a marker over the destination, and showing the description of buildings. The navigation will base on displaying arrows and directions in the real world through the camera.

The application will be intended for mobile devices with Android operating system with minimum version of Android 4.4 ‘KitKat’ and API level of 20. Such devices have already built-in camera, GPS and can handle AR technology, which is necessary for this project.

# 2. Software environment

In this section I will mention and describe all the environments and programs used to develop the app. Additionally I will compare them to other available solutions and explain why I decided to choose them.

## 2.1 Operating system: Android

The app will be implemented on the world’s most used mobile operating system – Android. Android is an operating system based on modified linux kernel and other open source software and according to the article from www.phonearena.com, “Google’s Android OS: Past, Present, and Future”[1] it was first released in 2008 on the HTC Dream G1 by company Android Inc. founded in 2003 and acquired by Google in 2005. As shown in the figure 1, in the year 2021 approximately 70% of sold mobile devices had Android operating system installed, and IOS being second with about 28% of the market share. Since the rest of the operating systems hold insignificant fraction of market share, the comparison will only focus on the first two.

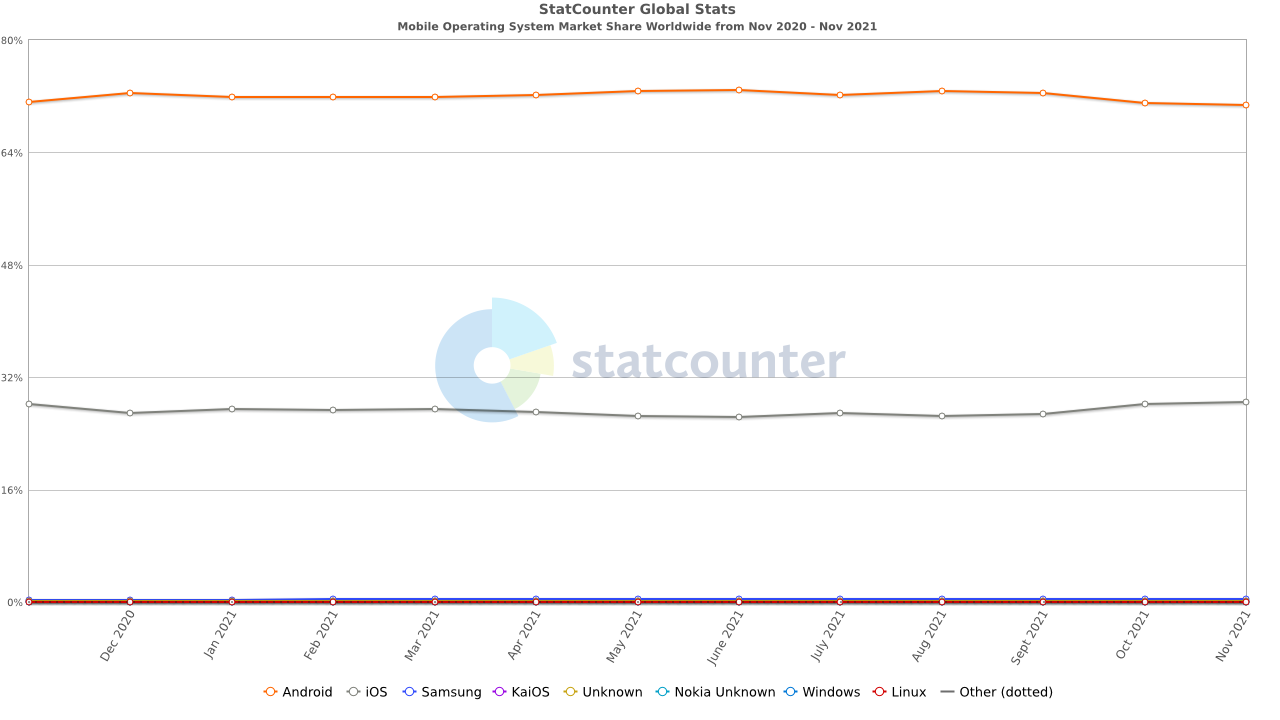


Figure 1 – Mobile operating systems market share

Android and IOS have separate stores for their available applications, Google Play for Android applications and Apple App Store for IOS applications. In terms of number of apps available on each store the Android Google Play is currently in the lead as can be seen on the chart in figure 2 with Apple App Store having about million less apps. Chart in figure 3 points out that more applications are downloaded from Google Play lately in comparison to IOS app store. It specifically shows data from third quarter of 2019 and 2020, where in the latter the difference was over 20 billion of downloaded apps. It also recorded much bigger increase in downloaded applications than Apple App Store in this time period.

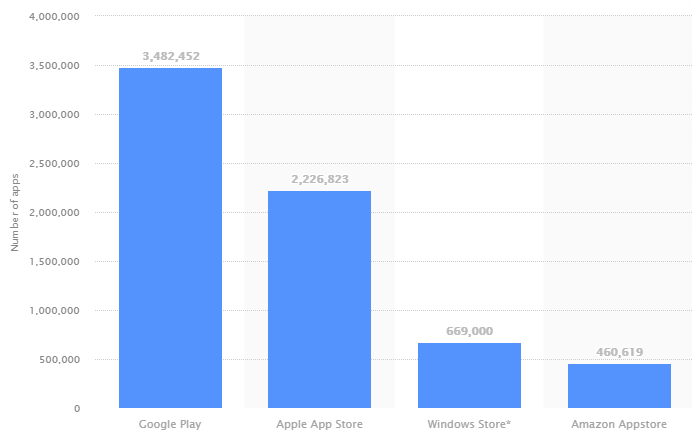


Figure 2 – Number of apps in each app stores

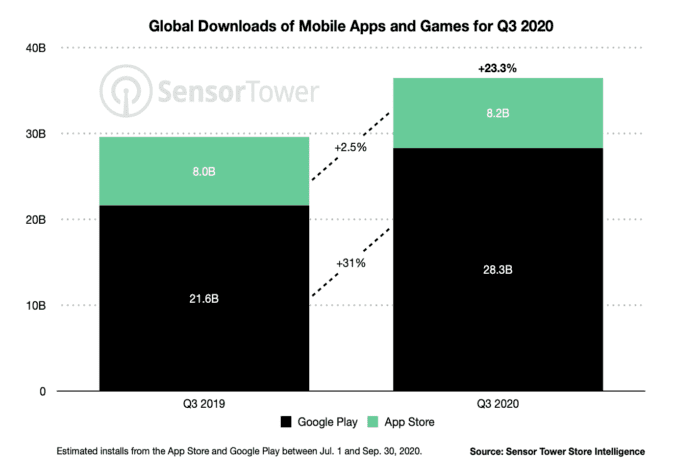


Figure 3 – Apps downloads comparison

According to the data presented above Android is by far the biggest mobile operating system and its application store is growing much faster than that of the second biggest operating system – IOS. This is the reasoning that caused choosing Android as intended operating system for application described in this work and because of that this application will be able to reach the most users.

# 2.2 Software environment: Unity

Unity is game engine running on Microsoft Windows, Linux and MacOS operating systems. It is designed to create three dimensional or two dimensional games or applications for different platforms like personal computers, game consoles or mobile devices as well as VR and AR content. As mentioned in the article by Eric Peckham on www.techcrunch.com, “How Unity built the world’s most popular game engine”[2] Unity was founded in Copenhagen by  Nicholas Francis, Joachim Ante, and David Helgason in 2002. As article by Marie Dealessandri “What is the best game engine: is Unity right for you?” says “[…] as of September 2019, 52% of the top 1,000 mobile games were powered by Unity, as well as 60% of all AR/VR content, according to the company. Unity game players are located in 195 countries -- which is literally every single country on the planet”. What makes this game engine so special that most of indie mobile developers use it? Marie Dealessandri explains further in her article that Unity has many advantages like its broad and differential Asset store, its speed and agility, it is easy to learn and free which attracts beginners, and most importantly for this work it is good for VR and AR developers. She also describes its downsides like it not being suitable for big projects, but they are irrelevant considering application development described in this work.

# Programming language

The source code of Unity game engine was mainly written in C++, whereas the game engine allows for writing scripts in UnityScript (similar to JavaScript), C# and Boo. In the AGH Guide application the only programming language used was C# due to its commonness and easy implementation.

# ARCore

Another important aspect to consider is the AR implementation tools. The best available tools are the ARKit by Apple and Google’s ARCore. They are open source tools that allow for creation of AR content for mobile devices. The first one is responsible for AR for devices with IOS operating system while the second one is for Android devices, and since the Android operating system has been chosen for this application, the Google AR Core is the only viable option. As Andrew Makarov writes in his article “ARKit vs ARCore: Image Detection and Tracking”[3] there are more devices with ARKit deployed than with ARCore enabled – as seen in figure 4 in 2020 there are almost twice as many ARKit devices and the users are significantly more active – however “[…] ARCore pulls ahead of ARKit when it comes to mapping. ARCore’s larger mapping dataset increases the speed and quality of mapping that is achieved through the collection and storage of 3D environment information”. This makes the choice of Google’s tool even stronger for this undertaking, although the ARKit is more popular.

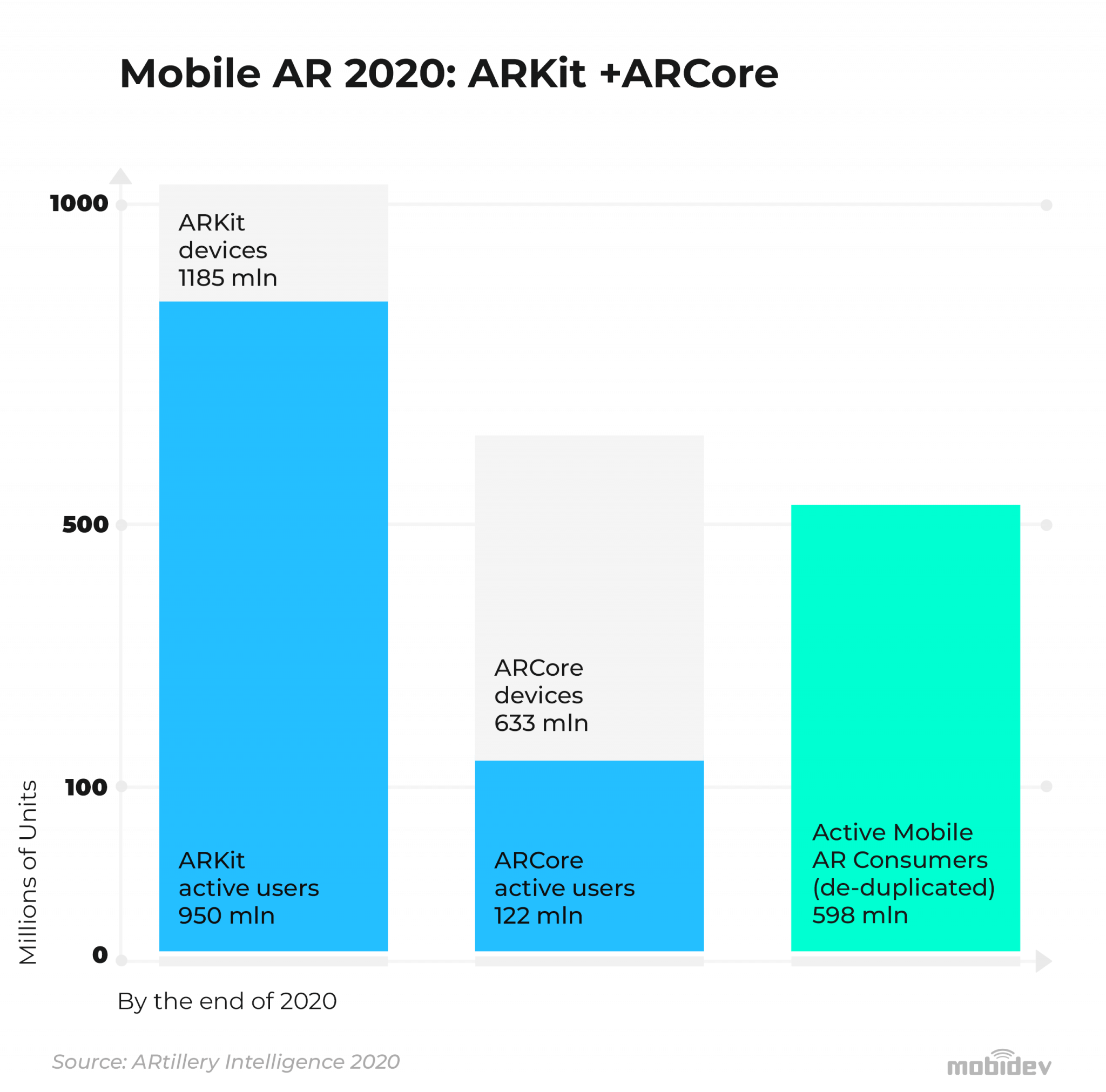


Figure 4 – Comparison of ARKit and ARCore numbers of devices

# Mapbox SDK

Mapbox SDK for unity is a software development kit created by Mapbox that allows creation of location based games, navigating applications or city simulators.

# 2.3 GIMP: graphics editor

Gimp is a free, open-source graphics editor mainly used for image editing or manipulation. This tool initially created as a semester-long project at University of California, Berkeley by  [Spencer Kimball](https://en.wikipedia.org/wiki/Spencer_Kimball_(computer_programmer)) and [Peter Mattis](https://en.wikipedia.org/wiki/Peter_Mattis) and its first release was in 1996. Its main advantages include simple interface, gradient editing, advanced manipulation and animation. It is also simple to use what makes it great for novice application and amateurs, so is perfectly sufficient for this case since it is used to create menu icons and simple graphics.

# 2.4 GitLab: git repository

To store the files and monitor different versions and changes in the application build GitLab web-based git repository was employed. To understand what GitLab is first one have to be familiar with what is Git – Git is a version control system used in computers to track any changes made in or to the files. It is mainly used to manage various changes in projects of any sizes and it also monitors the whole project. GitLab was developed and released in 2012 by Dimitriy Zaporozhets and Sid Sijbrandij, and is currently owned by GitLab Inc. It was one of the fastest-growing private company in 2018 in America. It is used by many different large and well-known organizations like Sony, NASA or IBM. It provides free, open or private repositories, manages changes made in the repositories and much more. For this project GitLab was used to create backup files, manage changes and monitor the whole application. It proved incredibly useful to store previous working versions of app with the ability to go back to them if something went wrong. In the figure 5 an exemplary uploaded application build is shown with ready and working navigation segment.

# References

Figure 1 - <https://gs.statcounter.com/os-market-share/mobile/worldwide>

Figure 2 – <https://www.statista.com/statistics/276623/number-of-apps-available-in-leading-app-stores/>

Figure 3 – <https://buildfire.com/app-statistics/>

Article in 2.1 – <https://www.phonearena.com/news/Googles-Android-OS-Past-Present-and-Future_id21273>

Article in 2.2 bout unity - <https://www.gamesindustry.biz/articles/2020-01-16-what-is-the-best-game-engine-is-unity-the-right-game-engine-for-you>

Article in 2.2 bout arkit and arcore - <https://arpost.co/2020/06/12/arkit-vs-arcore-image-detection-and-tracking/>

Figure 4 - <https://mobidev.biz/blog/augmented-reality-future-trends-2018-2020>