Czech University of Life Sciences Prague

Faculty of Economics and Management

Department of Information Technologies (FEM)

Bachelor Thesis

Cognitive Computing and Machine Learning

Fredy Alberto Rodriguez Bautista

CZECH UNIVERSITY OF LIFE SCIENCES PRAGUE

Faculty of Economics and Management

BACHELOR THESIS ASSIGNMENT

Fredy Alberto Rodriguez Bautista

Systems Engineering and Informatics
Informatics

Thesis title

Cognitive Computing and Machine Learning

Objectives of thesis

The main objective of the thesis is to find the best cloud cognitive service where the proposed app can be deployed.
Partial objectives are:
- to analyze current approaches,
- to test possible solutions if is suitable for the proposed objective,
- to develop an app, that through a camera(s) can identify and control the attendance, automatically and in real-time from a defined list of people.

Methodology

Thesis are based on scientific and proffesional literature. Theoretical part will contain research of available cloud cognitive solutions and their evaluation with respect to the aim of the thesis solved in the second (practical) part of the thesis.

The second part will consist of application of chosen cloud cognitive solution with the implement and development of an app that, through a camera, can detect the presence or attendance of the people linked to a organization list, their entry and exit time, and thus have a stricter attendance control.

The proposed extent of the thesis

40 – 50 pages

Keywords

Cognitive services, machine learning, cloud solutions, attendance sheet, attendance control

---

Recommended information sources

BISHOP, C M. *Pattern recognition and machine learning.* [New York]: Springer, 2006. ISBN 0-387-31073-8.

KONAR, A. *Artificial intelligence and soft computing : behavioral and cognitive modeling of the human brain.* LONDON: CRC, 2000. ISBN 0-8493-1385-6.

SUSSMAN, A. – HOLLANDER, J B. *Cognitive architecture : designing for how we respond to the built environment.* New York: Routledge, 2015. ISBN 978-0-415-72468-5.

UNPINGCO, J. *Python for probability, statistics, and machine learning.* New York, NY: Springer Berlin Heidelberg, 2016. ISBN 9783319307152.

VANDERPLAS, J T. *Python data science handbook : essential tools for working with data.* Beijing ; Boston ; Farnham ; Sebastopol ; Tokyo: O'Reilly, 2017. ISBN 978-1-4919-1205-8.

Expected date of thesis defence

2020/21 SS – FEM

The Bachelor Thesis Supervisor

Ing. Alexandr Vasilenko, Ph.D.

Supervising department

Department of Information Technologies

Electronic approval: 24. 11. 2020

doc. Ing. Jiří Vaněk, Ph.D.

Head of department

Electronic approval: 25. 11. 2020

Ing. Martin Pelikán, Ph.D.

Dean

Prague on 15. 03. 2021

Declaration

I declare that I have worked on my bachelor thesis titled "Cognitive Computing and Machine Learning" by myself and I have used only the sources mentioned at the end of the thesis. As the author of the bachelor thesis, I declare that the thesis does not break any copyrights.

In Prague on 15/03/2021 _____

Acknowledgement

First of all, I am grateful to God for the opportunity to study at this prestigious university. I would like to express my sincere gratitude especially to my supervisor Ing. Alexandr Vasilenko, Ph.D. for his continuous support, patience, motivation and immense knowledge. His guidance helped me all the time during the analysis and writing of this thesis. I am also thankful to all the teachers who shared their knowledge and advice during lessons, for their assistance, dedicated involvement and willingness during whole 3 years.

Cognitive Computing and Machine Learning

Abstract

The purpose of this work is to find the relation between the concepts of machine learning and cloud services, in order to develop an application that enables the automatic control of class or event attendance. To accomplish this, it was needed to study the concepts of biometrics, its relationship with machine learning and how these services are provided easily and accessible from the cloud through different providers, including Microsoft Azure, Amazon AWS and Google Cloud Platform. Consequently, a quantitative analysis of the aforementioned providers was performed to determine the one that best meets the objectives, especially in price, quality and development facilities. Finally, the application developed is presented, with all its capabilities, development, implementation stages and some examples of its operation.

Keywords: Machine Learning, Cloud Services, Microsoft Azure, Python, Streamlit, Cognitive Services, API, Biometrics

Cognitive Computing and Machine Learning

Abstrakt

Účelem této práce je najít vztah mezi koncepty strojového učení a cloudovými službami, aby bylo možné vyvinout aplikaci, která umožní automatickou kontrolu docházky jakékoli třídy nebo události.

K dosažení tohoto cíle bylo nejprve nutné prostudovat koncepty Biometrie, její vztah k strojovému učení a jak snadno jsou tyto služby poskytovány a přístupné z cloudu prostřednictvím různých poskytovatelů, včetně Microsoft Azure, Amazon AWS a Google Cloud Platform.

Následně byla provedena kvantitativní analýza výše uvedených poskytovatelů, aby se určil ten, který nejlépe vyhovuje cílům, zejména pokud jde o cenu, kvalitu a vývojová zařízení.

Nakonec je představena vyvinutá aplikace se všemi jejími schopnostmi, vývojovými a implementačními fázemi a několika příklady jejího fungování.

Klíčová slova: Machine Learning, Cloud Services, Microsoft Azure, Python, Streamlit, Cognitive Services, API, Biometrics

Table of content

List of Figure

List of tables

List of abbreviations

AI: Artificial Intelligence

ANN: Artificial Neural Network

API:

DNS: Domain Name System

DR: Detection Rate

HTML: Hyper Text Markup Language

HTTP: Hyper Text Transfer Protocol

HTTPS: Hyper Text Transfer Protocol Secure

K-NN: K-Nearest Neighbor

LR: Linear Regression

ML: Machine Learning

NPL, Neural Process Language

Pred.: Prediction

SQL: Structured Query Language

URL: Uniform Resource Locator

1 Introduction

This thesis will be dedicated to show how the cognitive services present in the cloud can be applied to real-life problems, deploying an application that is easy to operate and with greater scalability than using my own resources (Processor, GPU's, training models, etc).

Machine learning has come to make our lives easier, since a machine can emulate human senses, many of daily tasks can be possible automation scenario. In this work, we are going to see from the basic concepts of machine learning and cognitive services, how they are available in the cloud, how easily an application can be developed, and a useful for any company or establishment in order to control the attendance at an event or class.

The application developed allows the administrator user to take the attendance automatically in three different scenarios, manual, to analyse a single image, either using the camera of the device where the application was installed or uploading it, scheduled, where the time of the class and the attendance frequency is fixed by the user (e.g. SQL class from 8:00 am to 10:00 am each 2 minutes), and the completely automatic mode, where the application takes the attendance in an unlimited time with a fixed frequency (attendance each 2 minutes).

The language used for the development is python, the frontend is based on the Streamlit library, the API of the cognitive service used is AZURE. All the code is available on GitHub for your audit, feedback and improvement.

2 Objectives and Methodology

2.1 Objectives

2.1.1 Main Objective

The main objective of the thesis is to find the best cloud cognitive service where the proposed app can be deployed.

2.1.2 Partial Objectives

Partial objectives are:

- to analyse current approaches,
- to test possible solutions if is suitable for the proposed objective,
- to develop an app, that through a camera(s) can identify and control the attendance, automatically and in real-time from a defined list of people.

3 Methodology

The thesis is based on scientific and professional literature. The theoretical part will contain the research of available cloud cognitive solutions and their evaluation for the aim of the thesis solved in the second (practical) part of the thesis.

The second part will consist of an application of chosen cloud cognitive solution with the implementation and development of an app that, through a camera, can detect the presence or attendance of the people linked to an organization list, their entry and exit time, and thus have a stricter attendance control.

4 Literature Review

The literature of this thesis is divided into three major components connected. First I will find the field of application of the thesis, which is biometrics, secondly, I will demonstrate how machine learning (and artificial intelligence, deep learning) will help us to take advantage of the data generated by security cameras, which could become an ally of biometrics, and finally, see the platforms in the cloud that allow us to easily deploy a system of identification by cognitive vision, taking as input the biometric data sent through camera images, their processing, and its output as useful data, all without human intervention.

4.1 Biometrics and its importance in modern society

In ancient civilizations, people lived in small communities where they recognized each other without any difficulty, however, with large migratory movements and the exponential growth of the population, identification became a complex process, so that in modern societies it has been necessary. the implementation of sophisticated identity management systems, today we find from a simple identity document that makes you a member of a community, country, or group, to sophisticated facial recognition systems used by governments to control the population.

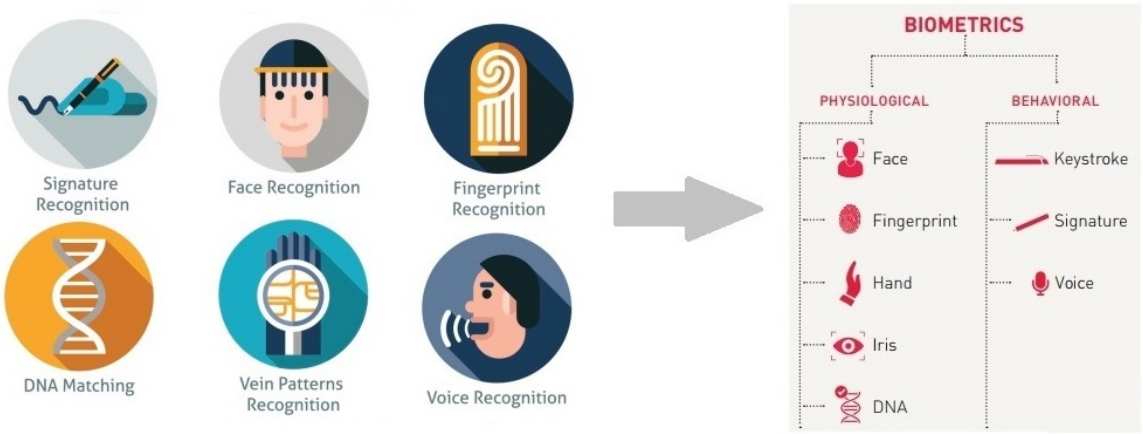
Identity, in this case, and during the development of this work, refers to the set of information associated with a person, such as a name, surname, date of birth, ID, entry, and exit movements, among others. Identity management systems are used in different applications such as customs, restricting and control access to facilities, controlling access to computer resources, in financial transactions, and boarding commercial flights, to name a few. Within the identity management systems are those that base their operation on biometrics that performs the analysis of the characteristics of the human body.

4.1.1 What is Biometric?

Biometrics in modern computer science is defined as the automated use of biological properties to identify individuals (Jiang, 2017). Biometrics are physical or behavioural human characteristics that can be used to digitally identify a person to grant access to systems, devices, or data. (Korolov, 2019).

Examples of these biometric identifiers are fingerprints, facial patterns, voice, or typing cadence (see Figure 1). Each of these identifiers is considered unique to the individual, and they may be used in combination to ensure greater accuracy of identification.

Figure 1 Types of Biometrics



Source: Thales Group (Thales Group, 2020)

4.1.2 Types of Biometrics?

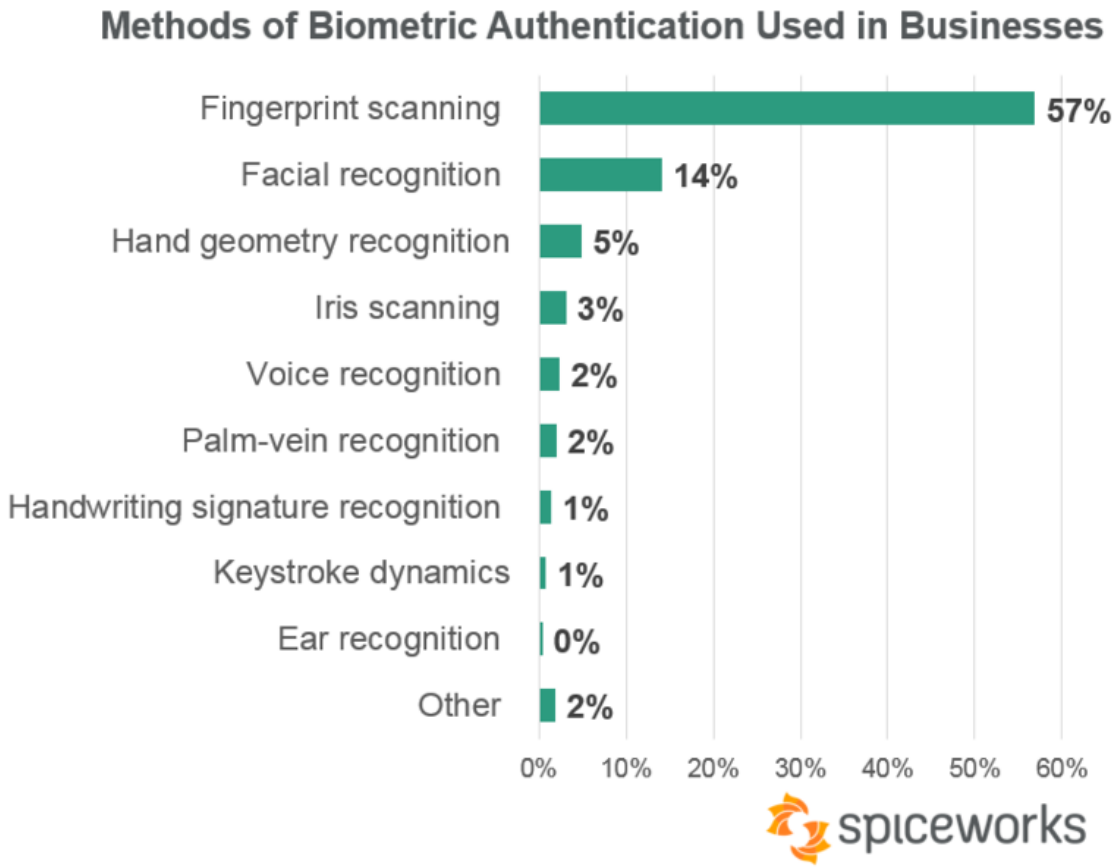
A biometric system aims to automatically identify or verify the identity of a user from the analysis of one or more features of the human body. Any person registered in said system is known as a user. They fall roughly into two categories: physical identifiers and behavioural identifiers. (Kaklauskas, 2015)

4.1.2.1 Physical Identifiers

A Physical identifier is, for the most part, immutable and device-independent:

Fingerprints: Fingerprint scanners have become ubiquitous in recent years due to their widespread deployment on smartphones. Any device that can be touched, such as a phone screen, computer mouse or touchpad, or a door panel, has the potential to become an easy and convenient fingerprint scanner.

Figure 2 Use Biometric authentication on Business Technologies



Source: (Tsai, 2018)

Physiological recognition: Facial recognition is the second most common type of authentication, according to Spiceworks[1] (Tsai, 2018), in place at 14 percent of companies. Other image-based authentication methods include hand geometry recognition, used by 5 percent of companies, iris or retinal scanning, palm vein recognition, and ear recognition (Tsai, 2018).

[1] Spiceworks is an online community where users can collaborate and seek advice from one another, and also engage in a marketplace to purchase IT-related services and products. The network is estimated to be used by more than six million IT professionals and 3,000 technology vendors.

The subject of this thesis will revolve around this type of recognition since it allows the identification of people en masse in the most repeated and least intrusive way. later the subject will be explained more in-depth, and its application in the real environment.

Voice: Voice-based digital assistants and telephone-based service portals are already using voice recognition to identify users and authenticate customers. According to Spiceworks, 2 percent of companies use voice recognition for authentication within the enterprise (Tsai, 2018).

Signature: Digital signature scanners are already in widespread use at retail checkouts and in banks and are a good choice for situations where users and customers are already expecting to have to sign their names.

DNA: Today, DNA scans are used primarily in law enforcement to identify suspects. In practice, DNA sequencing has been too slow for widespread use. This is starting to change. Last year, a $1,000 scanner hit the market that can do a DNA match at a minute and prices are likely to keep dropping[2].

4.1.2.2 Behavioural identifiers

Behavioural identifiers are a newer approach and are typically being used in conjunction with another method because of lower reliability. However, as technology improves, these behavioural identifiers may increase in prominence. Unlike physical identifiers, which are limited to a certain fixed set of human characteristics, the only limits to behavioural identifiers are the human imagination (Korolov, 2019).

These methods are very important nowadays since it allows companies to identify if the person accessing a system is real and not a robot. That can help a company filter out spam or detect attempts to brute-force a login and password. Here is the dilemma

[2] Developed at MIT Lincoln Laboratory, IdPrism and its award-winning algorithms provide rapid analysis for complex forensic DNA samples. (McGovern, 2019)

of this type of identification, where progress occurs at the same time, both in the detection of robots that want to access a system, and robots imitating human behaviour.

A clear example is seeming in the use of Google reCAPTCHA[3] version 3 of 2020, an advanced risk analysis engine, and adaptive challenges to keep malicious software from engaging in abusive activities on any website. Meanwhile, legitimate users will be able to log inn, make purchases, view pages, or create accounts and fake users will be blocked. all this without the need to complete any "challenge" with a photo (identify bicycles or boats) or write the text that many times is not readable by a human being.

Here are some common approaches:

Typing patterns: Everybody has a different typing style. The speed at which they type, the length of time it takes to go from one letter to another, the degree of impact on the keyboard.

Physical movements: The way that someone walks are unique to an individual and can be used to authenticate employees in a building, or as a secondary layer of authentication for particularly sensitive locations.

Navigation patterns: Mouse movements and finger movements on trackpads or touch-sensitive screens are unique to individuals and relatively easy to detect with software, no additional hardware is required.

Engagement patterns: We all interact with technology in different ways. How we open and use apps, how low we allow our battery to get, the locations and times of day we're most likely to use our devices, the way we navigate websites, how we tilt our phones when we hold them, or even how often we check our social media accounts are all potentially unique behavioural characteristics. These behaviour patterns can be

[3] reCAPTCHA v3 returns a score for each request without user friction. The score is based on interactions with your site and enables you to take an appropriate action for your site. (Google Developers, 2020)

used to distinguish people from bots until the bots get better at imitating humans. And they can also be used in combination with other authentication methods, or, if the technology improves enough, as standalone security measures (Korolov, 2019).

4.2 Cognitive and Machine Learning

Today in 2021 is very easy to find a surveillance camera anywhere. I remember as a child, in the mid-90s, the use of these devices was only reserved for financial institutions, they were very archaic and only had a certain range of vision, and their quality left much to be desired. Currently, camera surveillance systems are everywhere, they are so accessible in price, that, even a simple restaurant or bakery, can have 2 or 3 cameras, with a resolution higher than those of the best financial institutions in the 90s and they can also record in the dark.

However, of the petabits of data generated today, very few are processed. These cameras are only useful, in terms of security, if a person is 24 hours a day aware of what is happening live.

Here is where the topic of my thesis comes into play and is to make sense of the images taken by the cameras and based on a conjunction of artificial intelligence, machine learning, and deep learning, to detail the people present in a picture, either for their identification, its monitoring of entry and exit of a site, as a security method to give access to restricted sites, or control of attendance of a class.

4.2.1 Pattern Recognition

With the passage of time, human beings, like other organisms, have developed important mechanisms and skills to obtain information from the environment and make decisions based on it. For example, some birds evaluate the moment of carrying

out the migration to get better food sources or climates, they are able to identify their predators, they can mark and recognize territories, among others. The ability to recognize different situations and make pertinent decisions is an inherent capacity of living beings, apparently this action is seen as a simple common-sense task, however it involves a complex cognitive process.

The learning process in living beings has been the subject of interest in various areas such as psychology, psychosociology, biology, neuropsychology and information theory. On the other hand, exact sciences, computer science and engineering have been given the task of developing theories and techniques that attempt to emulate the cognitive process of living beings in order to implement them in intelligent systems and be useful in different domains of human activity.

Living beings obtain information from the environment through the senses such as sight, touch, taste, smell and hearing; while intelligent systems obtain it through sensors and present the data in a convenient way for processing and interpretation in computers. The recorded data are called patterns, and these can be represented as signals, images or data tables.

Pattern Recognition is a mature but exciting and fast developing field, which underpins developments in cognate fields such as computer vision, image processing, text and document analysis and neural networks. It is closely akin to machine learning, and also finds applications in fast emerging areas such as biometrics, bioinformatics, multimedia data analysis and most recently data science. (Pattern Recognition, 2020)

4.2.2 Machine Learning

Machine learning is the study of algorithms and statistical models, which is a subset of artificial intelligence. Systems use it to perform a task without explicit instructions and instead rely on patterns and inference. Thus, it applies to computer vision, software engineering, and pattern recognition.

Machine learning is done by computers with minimal assistance from software programmers. It uses data to make decisions and allows it to be used in interesting ways in a wide variety of industries. It can be classified as supervised learning, semi-supervised learning, and unsupervised learning. (Full Scale, 2019)

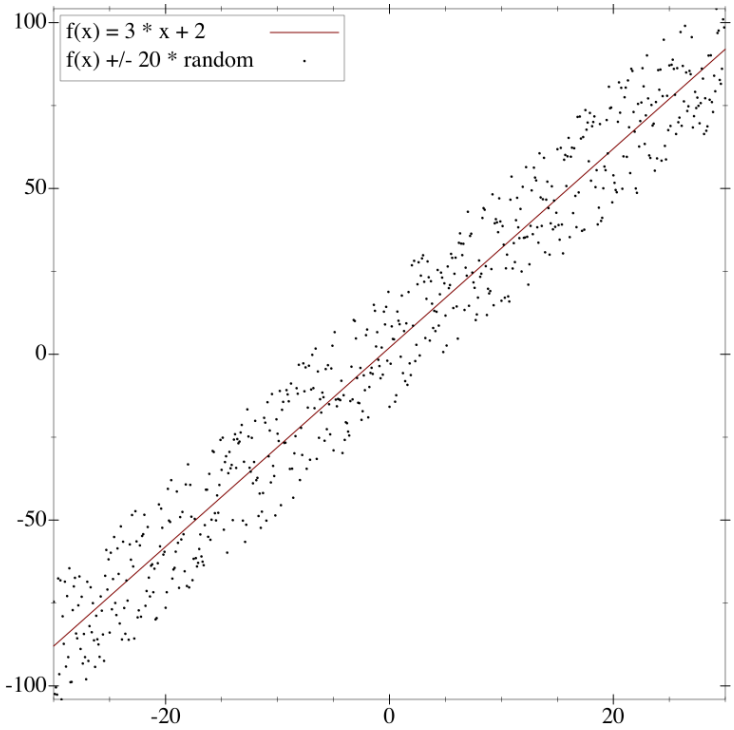
4.2.2.1 Supervised Learning

Supervised learning, or function approximation, is simply fitting data to a function of any variety (Kirk, 2017). For instance, given the noisy data shown in Figure 2, you can fit a line that generally approximates it. Those kinds of Machine Learning Algorithms where you use a collection of input-output pairs to make machines understand the relation between them. This collection of input-output pairs is called a labelled dataset. Applying that relation, the machine can then predict the output for a given unknown input. (Kalakheti, 2020)

Some of the examples of supervised Machine Learning Algorithms are:

- Linear Regression
- Artificial Neural Networks
- Logistic Regression
- Support Vector Machine
- Naive Bayes Classifier, etc.

Figure 3 This data fits quite well to a straight line



Source: Thoughtful Machine Learning with Python (Kirk, 2017)

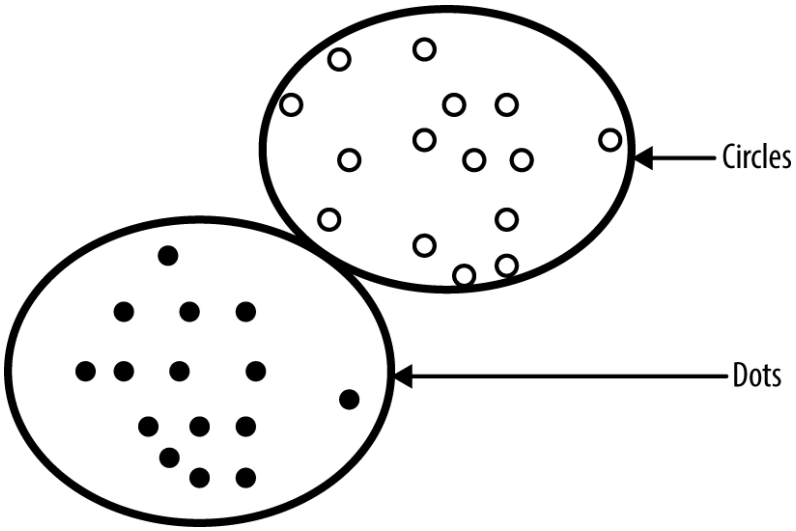
4.2.2.2 Unsupervised Learning

Unsupervised learning involves figuring out what makes the data special. For instance, if we were given many data points, we could group them by similarity (Figure 2), or perhaps determine which variables are better than others. “Those kind of machine Learning Algorithms where you have to use unlabelled dataset to train a machine learning model are unsupervised machine learning algorithms. They are mostly used for classification and clustering tasks. (Kalakheti, 2020)

Some of the examples of Unsupervised Machine Learning algorithms are:

- K-means clustering
- Autoencoders
- Principal Component Analysis
- Generative Adversarial Networks, etc.

Figure 4 Two clusters grouped by similarity

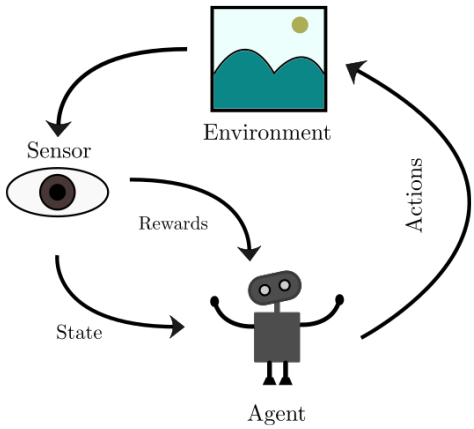


Source: Thoughtful Machine Learning with Python (Kirk, 2017)

4.2.2.3 Reinforcement Learning

Reinforcement learning involves figuring out how to play a multistage game with rewards and payoffs. Think of it as the algorithms that optimize the life of something. A common example of a reinforcement learning algorithm is a mouse trying to find cheese in a maze. For the most part, the mouse gets zero reward until it finally finds the cheese. (Kirk, 2017)

Figure 5 Basic of reinforcement Learning



Source: Intuition Behind Reinforcement Learning (García, 2019)

Other good example let's say you are learning cycling. Then, every time you fall, you realize that particular step used when you fell should not be used again i.e., punishment and every time you cycle without falling, you realize you should use that particular steps i.e., reward. After a few days of similar repetition, you will learn cycling. This is how reinforcement learning works. (Kalakheti, 2020)

4.2.3 Computer Vision

Computer vision is the process of understanding digital images and videos using computers. It seeks to automate tasks that human vision can achieve. This involves methods of acquiring, processing, analysing, and understanding digital images, and extraction of data from the real world to produce information. It also has sub-domains such as object recognition, video tracking, and motion estimation, thus having applications in medicine, navigation, and object modelling.

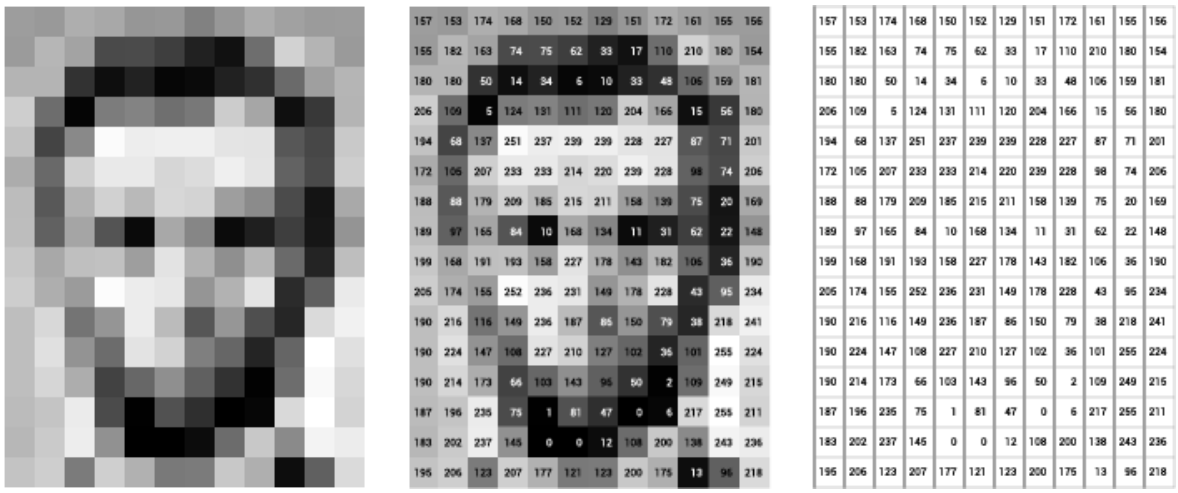
To put it simply, computer vision works with a device using a camera to take pictures or videos, then perform analysis. The goal of computer vision is to understand the content of digital images and videos. Furthermore, extract something useful and meaningful from these images and videos to solve varied problems. Such examples are systems that can check if there is any food inside the refrigerator, checking the health status of ornamental plants, and complex processes such as disaster retrieval operation.

4.2.3.1 How does computer vision work?

On a certain level Computer vision is all about pattern recognition. So, one way to train a computer how to understand visual data is to feed it images, lots of images thousands, millions if possible that have been labelled, and then subject those to various software techniques, or algorithms, that allow the computer to hunt down patterns in all the elements that relate to those labels.

Below is a simple illustration of the grayscale image buffer which stores our image of Abraham Lincoln. Each pixel's brightness is represented by a single 8-bit number, whose range is from 0 (black) to 255 (white). At left, our image of Lincoln; at centre, the pixels labelled with numbers from 0–255, representing their brightness; and at right, these numbers by themselves.

Figure 6 Pixel data diagram.



Source: Everything you ever wanted to know about computer vision. (Mihajlovic, 2019)

4.2.3.2 The relationship between Machine learning and computer vision

Technology never ceases to mimic the human brain; thus, AI gains a lot of interest for decades. To show the roadmap of these breakthroughs, let's discuss the relationship between AI, machine learning, and computer vision. AI is the umbrella of these fields, machine learning is a subset of AI, wherein computer vision is also the subset of machine learning. However, computer vision can be considered as a direct subset of AI.

Machine learning and computer vision are two fields that have become closely related to one another. Machine learning has improved computer vision about recognition and tracking. It offers effective methods for acquisition, image processing, and object focus which are used in computer vision. In turn, computer vision has broadened the

scope of machine learning. It involves a digital image or video, a sensing device, an interpreting device, and the interpretation stage. Machine learning is used in computer vision in the interpreting device and interpretation stage.

Relatively, machine learning is the broader field, and this is evident in the algorithms that can be applied to other fields. An example is the analysis of a digital recording, which is done with the use of machine learning principles. Computer vision, on the other hand, primarily deals with digital images and videos. Also, it has relationships in the fields of information engineering, physics, neurobiology, and signal processing.

The obstacle faced by developers and entrepreneurs is the huge gap between computer vision and biological vision. The fields most closely related to computer vision are image processing and image analysis. However, it deserves another interesting article to cite its relationship and differences. Also, the lack of knowledge about the main goal of machine learning in a particular project is a huge disruption among entrepreneurs.

4.3 Cloud Computing Services

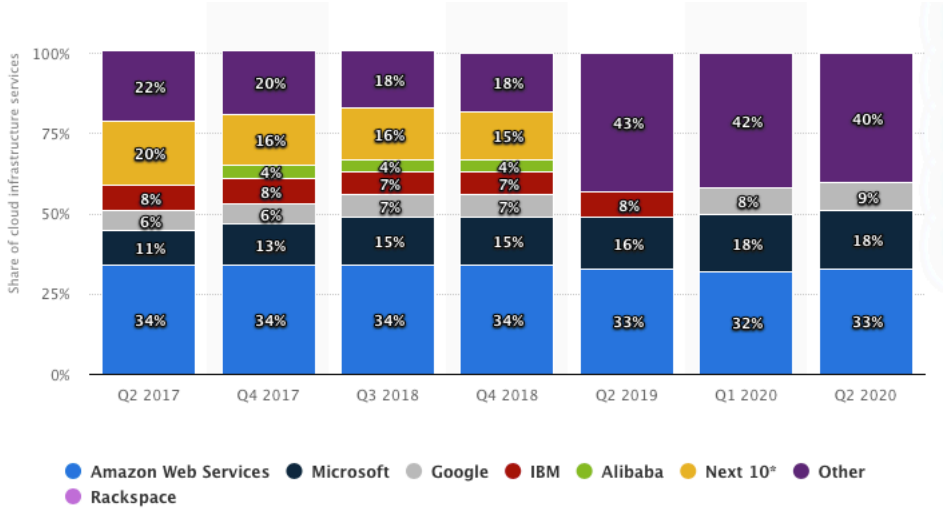
Cloud computing systems are a great innovation in the market, which means that from any computer or device with internet access they can have an almost unlimited amount of relocated and scalable resources according to needs. This is how, for example, from a raspberry pie 4[4] (a low-cost device with little powerful hardware) a complex machine learning algorithm can be run using a virtual machine installed on one of the servers of the cloud providers services, making use of the most powerful GPUs and processors on the market and having almost immediate results.

[4] Raspberry Pi is a series of small single-board computers developed in the United Kingdom by the Raspberry Pi Foundation in association with Broadcom. In its version 4 have CPU – Broadcom BCM2711, Quad core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz, RAM – 1GB, 2GB or 4GB LPDDR4-2400 SDRAM (depending on model) (RaspberryPi.org)

4.3.1 What is Cloud Computing?

Next, we will see how each of the largest cloud computing providers (according to the market share figure 7) on the market are defined themself.

Figure 7 Global market share of cloud infrastructure services from 2017 to 2020, by vendor



Source (Statista, 2021)

For Azure[5], simply put, cloud computing is the delivery of computing services—including servers, storage, databases, networking, software, analytics, and intelligence—over the Internet ("the cloud") to offer faster innovation, flexible resources, and economies of scale. You typically pay only for cloud services you use, helping you lower your operating costs, run your infrastructure more efficiently, and scale as your business needs change. (Microsoft , 2021).

For AWS[6], Cloud computing is the on-demand delivery of IT resources over the Internet with pay-as-you-go pricing. Instead of buying, owning, and maintaining physical data centres and servers, you can access technology services, such as computing power, storage,

[5] **Microsoft Azure**, commonly referred to as Azure, is a cloud computing service created by Microsoft for building, testing, deploying, and managing applications and services through Microsoft-managed data centres. It provides software as a service (SaaS), platform as a service (PaaS) and infrastructure as a service (IaaS) and supports many different programming languages, tools, and frameworks, including both Microsoft-specific and third-party software and systems (Microsoft , 2021).

[6] **Amazon Web Services** (AWS) is the world's most comprehensive and broadly adopted cloud platform, offering over 175 fully featured services from data centres globally (Amazon, 2021).

and databases, on an as-needed basis from a cloud provider like Amazon Web Services (AWS) (Amazon, 2021).

And in case of GCP[7], in cloud computing, the capital investment in building and maintaining data centres is replaced by consuming IT resources as an elastic, utility-like service from a cloud "provider" (including storage, computing, networking, data processing and analytics, application development, machine learning, and even fully managed services). (Google, 2021)

Whereas in the past cloud computing was considered the province of start-ups and aggressively visionary enterprise users, today, it is part of the enterprise computing mainstream across every industry, for organizations of any type and size.

4.3.2 Benefits of Cloud Computing

4.3.2.1 Cost

Cloud computing eliminates the capital expense of buying hardware and software and setting up and running on-site datacentres—the racks of servers, the round-the-clock electricity for power and cooling, and the IT experts for managing the infrastructure. It adds up fast. (Microsoft , 2021)

4.3.2.2 Global Scale

The benefits of cloud computing services include the ability to scale elastically. In cloud speak, that means delivering the right amount of IT resources—for example, more or less computing power, storage, bandwidth—right when they're needed, and from the right geographic location. (Microsoft , 2021)

[7] Google Cloud Platform, offered by Google, is a suite of cloud computing services that runs on the same infrastructure that Google uses internally for its end-user products, such as Google Search, Gmail, file storage, and YouTube. (Google, 2021)

4.3.2.3 Productivity

On-site datacentres typically require a lot of “racking and stacking”—hardware setup, software patching, and other time-consuming IT management chores. Cloud computing removes the need for many of these tasks, so IT teams can spend time on achieving more important business goals. (Microsoft , 2021)

4.3.2.4 Speed

Most cloud computing services are provided self service and on demand, so even vast amounts of computing resources can be provisioned in minutes, typically with just a few mouse clicks, giving businesses a lot of flexibility and taking the pressure off capacity planning. (Microsoft , 2021)

4.3.2.5 Performance

The biggest cloud computing services run on a worldwide network of secure datacentres, which are regularly upgraded to the latest generation of fast and efficient computing hardware. This offers several benefits over a single corporate datacentre, including reduced network latency for applications and greater economies of scale. (Microsoft , 2021)

4.3.2.6 Reliability

Cloud computing makes data backup, disaster recovery, and business continuity easier and less expensive because data can be mirrored at multiple redundant sites on the cloud provider’s network. (Microsoft , 2021)

4.3.2.7 Security

Many cloud providers offer a broad set of policies, technologies, and controls that strengthen your security posture overall, helping protect your data, apps, and infrastructure from potential threats. (Microsoft , 2021)

4.3.3 Types of Cloud Services

There are three major models of cloud computing services and they are known as Software-as-a-Service (SaaS), Platform-as-a-Service (PaaS), and Infrastructure-as-a-Service (IaaS). These cloud services may be offered in a public, private, or hybrid network (RAJIV CHOPRA, 2018).

4.3.3.1 Infrastructure-as-a-Service (IaaS)

The customer gets resources such as processing power, storage, network bandwidth, CPU, and power. Once the user gets the infrastructure, he controls the OS, data, applications, services, host-based security, and so on. (RAJIV CHOPRA, 2018)

4.3.3.2 Platform-as-a-Service (PaaS),

The customer is provided with the hardware infrastructure, network, and operating system to form a hosting environment. The user can install his applications and activate services from the hosting environment. (RAJIV CHOPRA, 2018)

4.3.3.3 Software-as-a-Service (SaaS),

The customer/user is provided access to an application. He has no control over the hardware, network, security, or OS (RAJIV CHOPRA, 2018)

4.3.4 Cloud Cognitive Services

Cognitive Services brings AI within reach of every developer—without requiring machine-learning expertise. All it takes is an API call to embed the ability to see, hear, speak, search, understand, and accelerate decision-making into your apps. Enable developers of all skill levels to easily add AI capabilities to their apps with modern application development.

The Cognitive Services are grouped (mainly) into five categories (Windows Developers, 2018):

- Vision—analyse images and videos for content and other useful information.
- Speech—tools to improve speech recognition and identify the speaker.
- Language—understanding sentences and intent rather than just words.
- Knowledge—tracks down research from scientific journals for you.
- Search—applies machine learning to web searches.

4.3.4.1 Cognitive Services Vision

I will focus here on computer vision, given that the cognitive service that we are going to use in the thesis. Although all providers differ in their services, we can summarize that the big three provide divide the vision service in:

- Computer Vision—Distil actionable information from images.
- Content Moderator—Automatically moderate text, images and videos for profanity and inappropriate content.
- Emotion—Analyse faces to detect a range of moods.
- Face—identify faces and similarities between faces.
- Video—analyse, edit and process videos within your app.

5 Practical Part

The practical part of the thesis will be divided into two, the first part of the practice will be to choose the cognitive service solution that best suits the project, for this, the three largest providers (Azure, AWS and GCP) will be compared, taking into account different factors such as price, stability, ease of use and implementation. The second practical part will deal with the development of the application and its deployment in the solution chosen in the previous point. Here the backend development of the application is included, the obtaining and pre-processing of the images, the connection and interaction of with the API, the storage of the results, among others. on the frontend side, the class attendance interface will be created (CRUD). The project will be carried out in Python version 3.

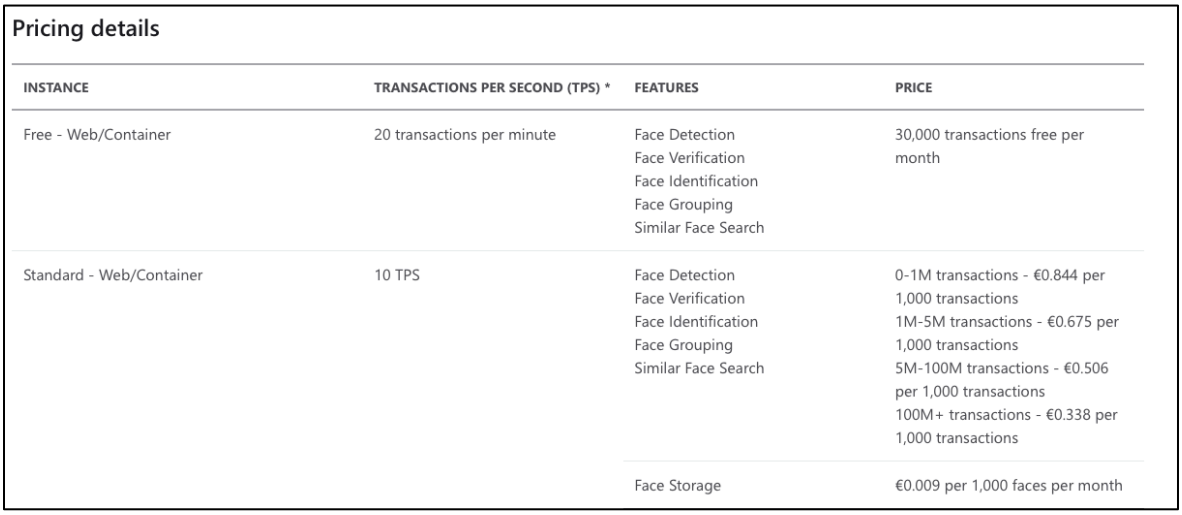
5.1 Cloud Service Selection

Each of the different applications have a different way of charging for their services, some are cheaper than others, however they offer a worse free plan, since in the first instance this is a school project, this topic will have a significant weight to the time of selection.

5.1.1 Face (Azure Solution)

Face is as AZURE name the solution for face recognition and is defined as Embed facial recognition into your apps for a seamless and highly secured user experience. No machine learning expertise is required. Features include face detection that perceives faces and attributes in an image; person identification that matches an individual in your private repository of up to 1 million people; perceived emotion recognition that detects a range of facial expressions like happiness, contempt, neutrality, and fear; and recognition and grouping of similar faces in images (Microsoft Azure, 2021).

Figure 8 Prices Face API January 2021



Pricing details

INSTANCE	TRANSACTIONS PER SECOND (TPS) *	FEATURES	PRICE
Free - Web/Container	20 transactions per minute	Face Detection Face Verification Face Identification Face Grouping Similar Face Search	30,000 transactions free per month
Standard - Web/Container	10 TPS	Face Detection Face Verification Face Identification Face Grouping Similar Face Search	0-1M transactions - €0.844 per 1,000 transactions 1M-5M transactions - €0.675 per 1,000 transactions 5M-100M transactions - €0.506 per 1,000 transactions 100M+ transactions - €0.338 per 1,000 transactions
		Face Storage	€0.009 per 1,000 faces per month

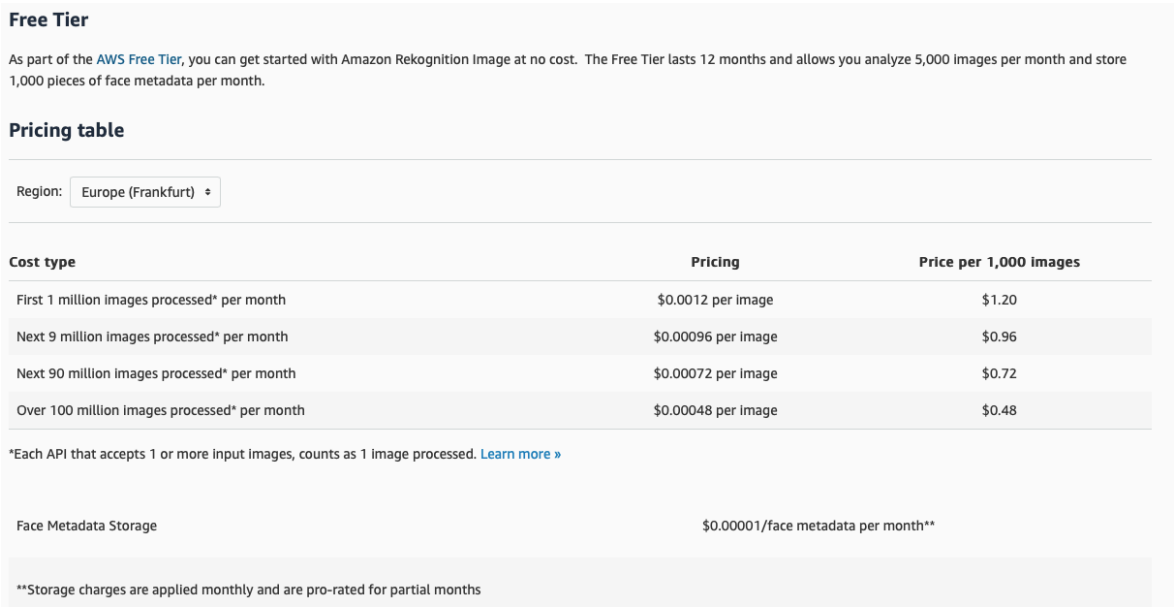
Source: (Microsoft Azure, 2021)

5.1.2 Rekognition (Amazon solution)

Amazon Rekognition makes it easy to add image and video analysis to your applications using proven, highly scalable, deep learning technology that requires no machine learning expertise to use. With Amazon Rekognition, you can identify objects, people, text, scenes, and activities in images and videos, as well as detect any inappropriate content. Amazon Rekognition also provides highly accurate facial analysis and facial search capabilities that you can use to detect, analyse, and compare faces for a wide variety of user verification, people counting, and public safety use cases.

With Amazon Rekognition Custom Labels, you can identify the objects and scenes in images that are specific to your business needs. For example, you can build a model to classify specific machine parts on your assembly line or to detect unhealthy plants. Amazon Rekognition Custom Labels takes care of the heavy lifting of model development for you, so no machine learning experience is required. You simply need to supply images of objects or scenes you want to identify, and the service handles the rest.

Figure 9 Amazon Rekognition Service Price January 2021.



Free Tier

As part of the AWS Free Tier, you can get started with Amazon Rekognition Image at no cost. The Free Tier lasts 12 months and allows you analyze 5,000 images per month and store 1,000 pieces of face metadata per month.

Pricing table

Region: Europe (Frankfurt)

Cost type	Pricing	Price per 1,000 images
First 1 million images processed* per month	$0.0012 per image	$1.20
Next 9 million images processed* per month	$0.00096 per image	$0.96
Next 90 million images processed* per month	$0.00072 per image	$0.72
Over 100 million images processed* per month	$0.00048 per image	$0.48

*Each API that accepts 1 or more input images, counts as 1 image processed. Learn more »

Face Metadata Storage	$0.00001/face metadata per month**

**Storage charges are applied monthly and are pro-rated for partial months

Source: (Amazon Rekognition, 2021)

5.1.3 Vision AI (Google Solution)

Google Cloud's Vision API offers powerful pre-trained machine learning models through REST and RPC APIs. Assign labels to images and quickly classify them into millions of predefined categories. Detect objects and faces, read printed and handwritten text, and build valuable metadata into your image catalogue.

Figure 10 Price Vision AI January 2021

Feature	Price per 1000 units		
	First 1000 units/month	Units 1001 - 5,000,000 / month	Units 5,000,001 and higher / month
Label Detection	Free	$1.50	$1.00
Text Detection	Free	$1.50	$0.60
Document Text Detection	Free	$1.50	$0.60
Safe Search (explicit content) Detection	Free	Free with Label Detection, or $1.50	Free with Label Detection, or $0.60
Facial Detection	Free	$1.50	$0.60
Facial Detection - Celebrity Recognition	Free	$1.50	$0.60
Landmark Detection	Free	$1.50	$0.60
Logo Detection	Free	$1.50	$0.60
Image Properties	Free	$1.50	$0.60
Crop Hints	Free	Free with Image Properties, or $1.50	Free with Image Properties, or $0.60
Web Detection	Free	$3.50	Contact Google for more information
Object Localization	Free	$2.25	$1.50

Source: (Google Cloud, 2021)

5.1.4 Cost Analysis

During the cost selection it was found that the service that offers the best price for both large-scale solutions and specific products is azure.

Table 1 Cost per Transactions per Solution

Transactions	Azure (Free)	Azure(S1)	AWS	GCP
1000	€ 0.00	€ 0.00	€ 0.00	€ 0.00
2000	€ 0.00	€ 0.00	€ 2.40	€ 3.00
3000	€ 0.00	€ 0.00	€ 3.60	€ 4.50
5000	€ 0.00	€ 0.00	€ 6.00	€ 7.50
10000	€ 0.00	€ 8.40	€ 12.00	€ 15.00
100000	Use S1	€ 84.00	€ 120.00	€ 150.00

Source: Own elaboration.

In Table 1 we can see the cost of each platform, a transaction is called any request made to the platform, such as detecting the people in each image, the training of the recognition model, creating a new group, etc. To highlight, the azure service offers two types of services, one free (for 12 months) where 30,000 requests to the platform are free each month. the other service has 5000 free transactions.

According to Table 1, it is clear that the cheapest service (January 2021) is offered by Azure, either for a small implementation (on what this thesis is focused on) and for larger solutions, details that will be addressed below.

As the purpose of this thesis in the development of an application where class attendance is automated, we are going to evaluate the costs for a classroom (30 people), taking attendance every 10 minutes, during a standard class of 90 minutes.

Table 2 Price per a 90 mins class per Platform

Class Scheduled	Transactions per class*	Azure (Free)	Azure(S1)	AWS	GCP
1	270	€ 0.00	€ 0.00	€ 0.00	€ 0.00
10	2700	€ 0.00	€ 0.00	€ 3.24	€ 4.05
100	27000	€ 0.00	€ 22.68	€ 32.40	€ 40.50
1000	270000	Use S1	€ 226.80	€ 324.00	€ 405.00
10000	2700000	Use S1	€ 2,268.00	€ 3,240.00	€ 4,050.00

Source: Own Elaboration.

$$* Transaction\ per\ class = Student\ per\ class * \frac{Class\ Lenght\ in\ minutes}{10\ minutes}$$

$$Transaction\ per\ class = 30 * \frac{90}{10}$$

$$Transaction\ per\ class = 270$$

Each new user represents a cost, and this will be quantified in the following table. Based on the thesis, each user must upload to the platform between 3 and 5 photographs (5 will be taken for the price calculation) from different angles, in order to have a solid model and with great confidence.

Table 3 Price Model Training per User

Users	Transaction	Azure (Free)	Azure(S1)	AWS	GCP
1	5	€ 0.00	€ 0.00	€ 0.00	€ 0.00
100	500	€ 0.00	€ 0.00	€ 0.00	€ 0.00
1000	5000	€ 0.00	€ 4.20	€ 6.00	€ 7.50
5000	25000	€ 0.00	€ 21.00	€ 30.00	€ 37.50
10000	50000	Use S1	€ 42.00	€ 60.00	€ 75.00
100000	500000	Use S1	€ 420.00	€ 600.00	€ 750.00

Source: Own elaboration

Another cost, although marginal in the solution, is the storage of user metadata, which can be seen in table 4.

Table 4 Storage Price per user

Storage	Azure	AWS	GCP
1000	€ 0.01	€ 0.01	€ 0.01
2000	€ 0.02	€ 0.02	€ 0.02
3000	€ 0.03	€ 0.03	€ 0.03
5000	€ 0.05	€ 0.05	€ 0.05
10000	€ 0.09	€ 0.10	€ 0.10
100000	€ 0.90	€ 1.00	€ 1.00

Source: Own Elaboration

To conclude the cost analysis, we are going to simulate the cost per month of use per student, for this we assume that each student takes 6 classes per semester. Costs include storage and training of the model.

Table 5 Cost User per Month

Users	Subjects	Transaction per User	Azure Free	Azure(S1)	AWS	GCP
1	6	221	0	€ 0.00	€ 0.00	€ 0.00
100	600	22100	0	€ 18.56	€ 26.52	€ 33.15
1000	6000	221000	Use S1	€ 185.64	€ 265.20	€ 331.50
5000	30000	1105000	Use S1	€ 928.20	€ 1,326.00	€ 1,657.50
10000	60000	2210000	Use S1	€ 1,856.40	€ 2,652.00	€ 3,315.00
100000	600000	22100000	Use S1	€ 18,564.00	€ 26,520.00	€ 33,150.00

Source: Own Elaboration

Analysing the data previously obtained, we can confirm that the platform that has the best price at this time (January 2021) for the development of the application is Azure, both, in the free version, which would be enough for about 100 students per semester (table 5), as in the paid version. In second place we have AWS, approximately 43% more expensive than Azure, which does not offer any free service (only 1000 transactions per month) and finally we have GCP, 80% more expensive per transaction and like AWS, does not offer a version free.

5.1.5 Development analysis

This part will be evaluated according to the support offered in Python for the development of the application. It will search if the APIs have support and example codes for this language, as well as the community and the support given by the platform.

Table 6 Development Analysis

Item	Azure	AWS	GCP
Python Support	Yes	Yes	Yes
Developer Guide	Yes	Yes	Yes
Clear Documentation	Yes	Yes	Yes
Facial Recognition	Yes	Yes	No
Free Training	Yes	No	No

All Examples in Python	No	Yes	Yes
Live Support	No	Yes	No
Support by email	Yes	Yes	Yes
Result	**75%**	**88%**	**63%**

Source: Own Elaboration

Although it is a subjective analysis given that it is based on the research and how I feel with each of the platforms, and also that all the platforms resemble each other, I want to highlight AWS for having all the examples in Python as its main language, in Azure I highlight the free trainings with which you can make use of the models and in a negative way I mention GCP, since currently it no longer provides the individual facial recognition service on the platform, something that unquestionably takes it out of the current options .

5.1.6 Service Level agreement

Although the three platforms have an SLA close to 100% (Table 7), if there is an element that varies and that we are going to analyse in this section, it is how each platform responds to failures and how it repairs the damage caused in the presence of the service.

Table 7 SLA Vision services

SLA	Azure	AWS	GCP
Vision Services	99.90%	99.90%	99.90%

Source Own elaboration.

5.1.6.1 Face (Azure)

"Monthly Uptime Percentage" for each API Service is calculated as Total Transaction Attempts Less Failed Transactions divided by Total Transaction Attempts in a billing month for a given API subscription. Monthly Uptime Percentage is represented by the following formula:

Monthly Uptime % = (Total Transaction Attempts - Failed Transactions) / Total Transaction Attempts * 100

The following Service Levels and Service Credits are applicable to Cognitive Services APIs:

Table 8 Service Credit Azure

MONTHLY UPTIME PERCENTAGE	SERVICE CREDIT
< 99.9%	10%
< 99%	25%

Source: Own Elaboration

5.1.6.2 Rekognition (AWS)

Service Credits are calculated as a percentage of the total charges paid by you (excluding one-time payments such as upfront payments made for Reserved Instances) for the individual Included Service in the affected AWS region for the monthly billing cycle in which the Unavailability occurred in accordance with the schedule below.

Table 9 Service Credit AWS

MONTHLY UPTIME PERCENTAGE	SERVICE CREDIT
Less than 99.99% but equal to or greater than 99.0%	10%
Less than 99.0% but equal to or greater than 95.0%	30%
Less than 95.0%	100%

Source: Own Elaboration

5.1.6.3 Vision (GCP)

During the Term of the agreement under which Google has agreed to provide Google Cloud Platform to Customer, the Covered Service will provide a Monthly Uptime Percentage to Customer as follows:

Table 10 Service Credit GCP

MONTHLY UPTIME PERCENTAGE	SERVICE CREDIT

99% – < 99.9%	10%
95% – < 99.0%	25%
< 95%	50%

Source: Own Elaboration

5.1.7 Final selection

As mentioned before, the cost of services will be the item with the greatest weight in the selection (60%) accompanied by how easy it can be to develop in the selected platform (30%) and lastly the SLA of each platform and how it responds to any failure (10%).

Table 11 Weighted Selection Score

	Weight	Azure	AWS	GCP
Cost	60%	10	7	6
Development	30%	7	9	3[8]
SLA[9]	10%	8	10	9
Total	**100%**	**8.9**	**7.9**	**5.4**

Source: Own Elaboration

Unquestionably the winner and the platform that according to the numbers is the one indicated for the deployment of the application is Azure, its price is the most competitive in the market, also its free version is very wide (32000 transactions per month), which allows the development at almost zero cost. It has good documentation (There is no one better than Microsoft to write this) as well as clear examples (although the use of C # prevails). as for the rest evaluated, it is within the figures of the other competitors.

In second place we have AWS, although its price is up to 43% more expensive, its platform is more friendly for development, in addition to having an SLA policy that can return up to 100% of the credits consumed.

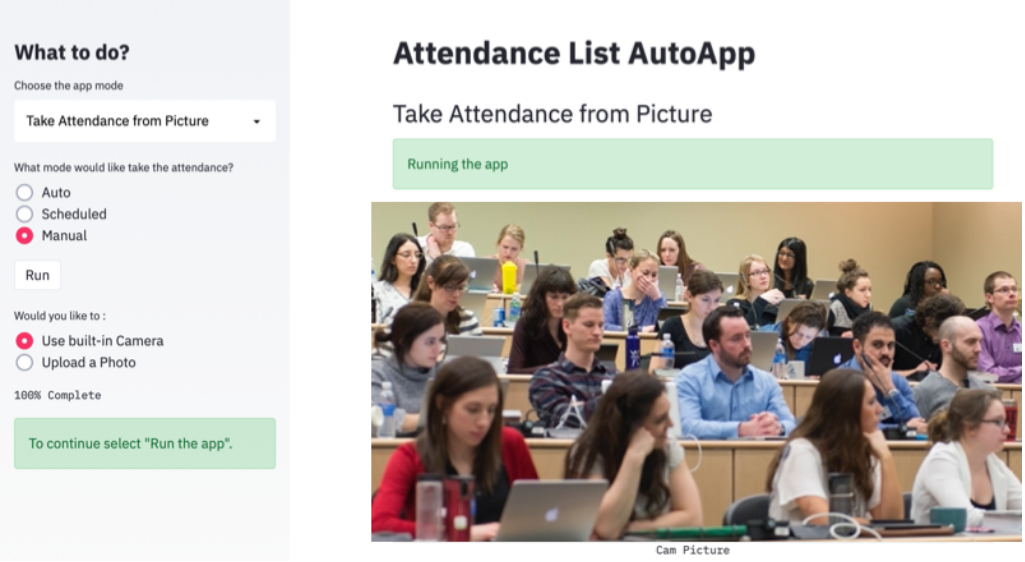
[8] A low rating has been given on this item as GCP does not allow facial recognition for the purpose of this project.
[9] Although the SLA is the same, the way the platform compensates is different

In the last place we have Google, its platform is very good also, however, since this year the individual facial recognition service is not supported, making the easy development of the application impossible, and its price is up to 80% higher than what offered by Azure.

5.2 Application development

Attendance List AutoApp is an app developed in Python and Streamlit[10] that provides fast, flexible, and automatically the attendance taking of any event or class, although its use can be extended to companies or any organization that needs to control the entry and exit hours of its employees, the detection of unauthorized personnel in any facility or serve as an access key that only allows the door to be opened if the person's face is registered.

Figure 11 Webapp Interface



Source: own elaboration

It aims was the implementation of the services provided by azure (platform chosen in the previous chapter) in a graphic way and easy to use for any user. The code is open and can be accessed from GitHub, However, I will be kept the project private until the presentation of this work, for two main reasons, the first one, the code can be copied

[10] Streamlit is an open-source Python library that makes it easy to create and share beautiful, custom web apps for machine learning and data science.

and later detected as plagiarism, and second, the keys that connect the application with the azure API are visible and can generate costs charged to my account for misuse.

If you require access to my code before its publication, please write to me at frealroba@gmail.com, and I will give you access so that you can download the code and use the connection keys to the Azure API for your evaluation and feedback.

After the evaluation and defence of the thesis, anyone is free to use my code, they just have to open an account in Azure and request their own keys (https://azure.microsoft.com/en-us/).

5.2.1 Main Features

Here are just a few of the things that application does well:

- **Current Groups,** here the user will find the registered groups. A group can be teachers, students, administrative employees, statistics II, etc.
- **Create a New Group**, as the name implies, there is a new group where you can add different members.
- **Add a New Member** is one of the key tasks of this application, first you have to identify the code of the new member, the application verifies if it was previously registered. If the code was not registered before, the program requests three photos of the new member's face (right side, front and left side) so that the machine learning model in the cloud is trained and, in this way, recognizes the person registered.

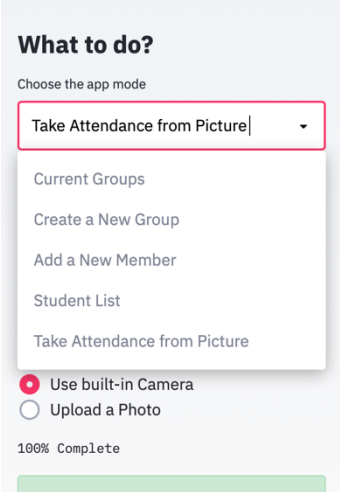


Figure 12 App Features.

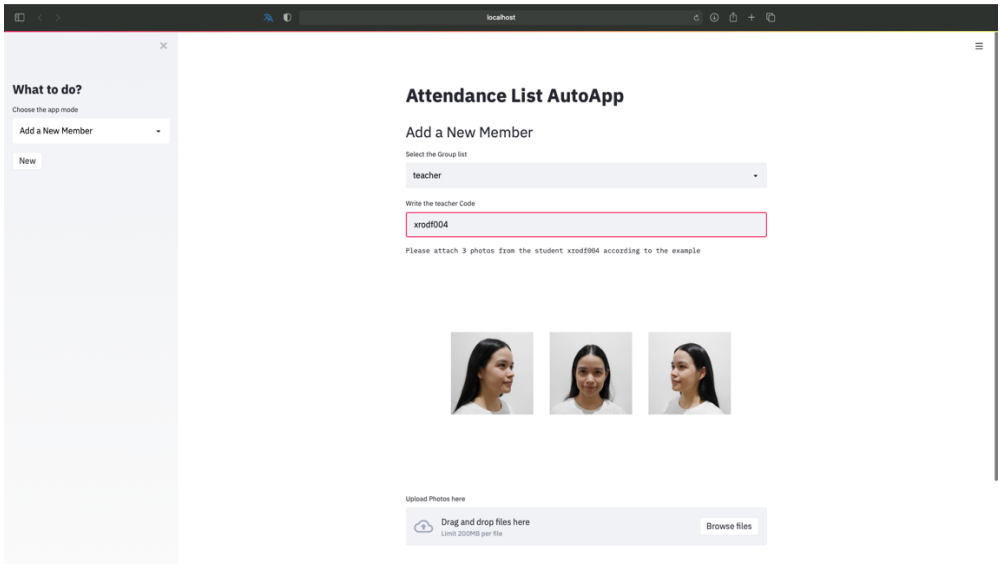
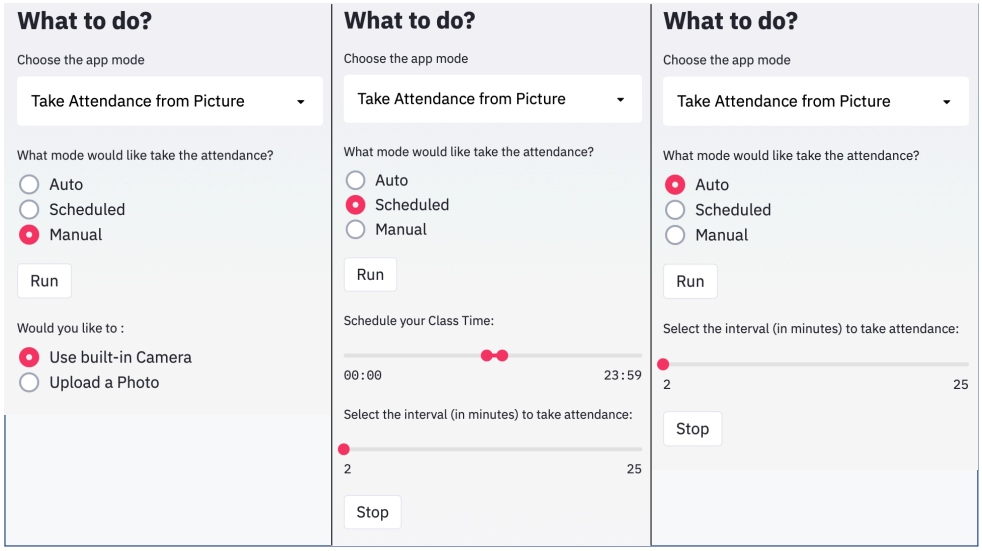


Figure 13 Add new member.

- **Student list** shows the students registered under a group.
- **Take attendance from a picture.** The application core and the most developed part. Here the user has three different options that allow covering a wide range of scenarios. Auto, Scheduled and Manual.

Figure 14 Attendance modes



Manual: User can take the assistance with a single photo, either using the camera of the device where the code is running or uploading a photo.

Scheduled: User can define the time and how often the program runs.

Auto: User define how often the attendance is taking. The program runs unlimitedly, and it will only end until the stop button is pressed.

5.2.2 Where to get it

The source code is currently hosted on GitHub at https://github.com/ingfrerod/Thesis , this is private repository. As mentioned at the beginning, the code will be public after the defence of the thesis.

5.2.3 Dependencies

In addition to Python, azure, and Streamlit, many other libraries were used, all of them are detailed in an attached file called requirements.txt.

5.2.3.1 Python

In addition to being one of the most popular languages in machine learning, I have experience with it. for this reason, is used in this project. The recommended version is python 3.7 or above.

5.2.3.2 Streamlit

Streamlit is the best library to launch a machine learning web application. It is flexible and allows to have an acceptable frontend without spending a lot of time in its development. This library was chosen as this project gives priority to backend and machine learning over web programming. Recommended version is 0.075.0

5.2.3.3 Azure Cognitive service API

API documentation can be easily consulted in here https://docs.microsoft.com/en-gb/azure/cognitive-services/face/ .

The following classes and interfaces handle some of the major features of the Face Python client library:
FaceClient: This class represents your authorization to use the Face service, and this is needed for all Face functionalities.
FaceOperations: This class handles the basic detection and recognition tasks that you can do with human faces.

DetectedFace: This class represents all of the data that was detected from a single face in an image. You can use it to retrieve detailed information about the face.
FaceListOperations: This class manages the cloud stored FaceList constructs, which store an assorted set of faces.
PersonGroupPersonOperations: This class manages the cloud-stored Person constructs, which store a set of faces that belong to a single person.
PersonGroupOperations: This class manages the cloud stored PersonGroup constructs, which store a set of assorted Person objects.

5.2.4 Installation

The application can be run on any operating system that allows python to run. It has been tested on MacOS (Big Sur, Catalina, Mojave), Windows (windows 10) and Linux (Ubuntu 20.10). Basic requirement is Python 3.7 (recommended) or higher.

Procedure in Linux/Mac terminal and Windows CMD:

- Clone the repo:

  $ git clone https://github.com/ingfrerod/Thesis

  Or download the ZIP file.

- Change the working directory to Thesis

  E.g., cd Thesis

- Install the requirements

  python3 -m pip install -r requirements.txt

  or

  python -m pip install -r requirements.txt

5.2.5 Running the App

Once installed, navigate to the working directory "Thesis" in the terminal or CMD, and execute the following line

Streamlit run main.py

Password : pwd123

Accept the permissions to use the device's camera and you will be able to use the program. Any problem at the time of installation or execution please consult it at frealroba@gmail.com.

5.2.6 Usage Example

The Covid pandemic of 2020 and 2021 has affected the application use tests, therefore tests have been carried out with family photos where the use of the program is emulated.

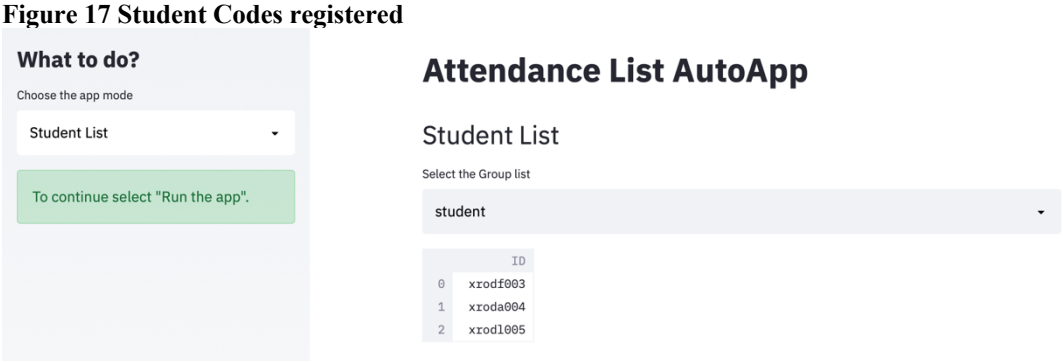
Figure 15 Model Training photos

The program has been trained with three people for recognition in any photo. Each person has uploaded three face photos as seen in figure 15 and given your student code as the only identifier.

Figure 16 Test Successful

We have passed through the model a photo where several people are, including the three faces with which the model trained. After executing the code, we have as output figure 16, where the 'Students' registered in the previous point are identified. This we can contrast with figure 17, where is the list of registered codes.

Figure 17 Student Codes registered



5.2.7 Future developments

The application from my point of view can be developed as a product and be attractive to any company. I would like to further refine the model making it more robust.

In this work, cloud services were used for the development of the application, in the future and for my master's thesis, I am going to extend the development to the use of my own machine learning models (with their pros and cons) achieving a product versatile that can be used both, online and offline.

6 Results and Discussion

The results of the work will be evaluated according to the objectives set. I will start with the partial's objectives, since this will facilitate us in the end to give an answer and see if the main objective was satisfactorily met.

6.1 Analysis of Partial Objectives and Results

6.1.1 Analyse current approaches

This objective was fully covered in the literary review, where I express the current state of cloud services, how they make use of machine learning, their connection with biometrics, the main providers in the market, their solutions offered and how they develop their product.

6.1.2 Test possible solutions if is suitable for the proposed objective

This objective is fulfilled in the practical part, where the service offered for main cloud service providers (Microsoft Azure, Amazon AWS and Google Cloud Platform), their prices, their facilities for the implementation of a solution and their level of service was meticulously verified and analysed. As a result, Azure was the winner, as it combines a great price, a stable platform, and a nearly 100% service level. In addition, its free trial was the one that best suited the work of this thesis, allowing the incorporation of almost 100 users for free (30,000 transactions per month in its S1 Plan).

One thing highlighted here is that GCP was practically disqualified, since in 2020 they had withdrawn their facial recognition model.

6.1.3 Develop an app, that through a camera(s) can identify and control the attendance, automatically and in real-time from a defined list of people.

This accomplished objective fills me even more with pride, since here I have developed a functional product that can be attractive to companies. My goal is to mature the idea and possibly create access control solutions for companies. The application developed uses of the device's camera where it was installed to automatically take the attendance of a class or event. It was developed in the backend

with Python and its frontend is based on a library of the same language called Streamlit. App use of Azure cognitive services for training the model and its subsequent identification of faces in a photograph.

The code is open for use, and above all for the feedback of the interested community, however, the keys that connect the application with Azure are for private use.

6.2 Analysis of main Objective and Results

The main objective of the thesis is to find the best cloud cognitive service where the proposed app can be deployed.

Having already analysed the partial objectives, we can determine that the goal of this other work was satisfactorily fulfilled, first we learned about the great world of biometrics, machine learning and its relationship with the services offered in the cloud, then we determined the best service offered in the moment of development of this thesis based on the price, its facilities to implement a solution and its quality of service offered, and finally, we have the development of the application.

6.3 Discussion

Cognitive cloud services are very convenient for a quick development and gives us almost unlimited growth power. As I have said, in my concept this application has a future as a profitable business idea. For this reason, my decision is to continue with its development, and for a future master's thesis, presenting a variable that makes use of machine learning models directly developed and trained by myself, this is a very big step that has to be taken. Although the cloud cognitive services give us the advantages mentioned above, in the long term the cost will be high, also the devices must be connected to the internet all the time and that is not possible in all cases. A hybrid solution is the next step in this work and what I will focus on in the future.

Privacy is a topic that was not addressed during the work, since it is out of our scope, and of course, it will be taken into account for future developments and start-up of the project.

7 Conclusion

In the previous chapter, we compare the results versus the proposed objectives, here we will be a little more pragmatic and we will conclude what we learned in the research and development of the application. For this reason, we highlight as conclusions:

- With quantitative analysis, we were able to reach the conclusion that the best Cognitive services for deploy the attendance list app is Microsoft Azure, At least in January 2021 Azure is 43% cheaper than AWS and almost 80% cheaper than Google Cloud platform. Also, Azure Offers a free service useful for around 100 users (30.000 transaction per month per 12 months)
- Cloud Computing services is the fast way to create machine learning applications, eliminates the capital expense of buying hardware (powerful processors ang GPUs) and software, also, no necessary setting up and running on-site data centres (servers, electricity, IT experts for managing the infrastructure)
- Has the ability to scale elastically, more specially, vast amounts of computing resources can be provisioned in minutes according to the service demand.
- The application developed is in a beta phase and has a lot of potential as a product for companies or any organization that wants to control its employees.
- Privacy is an important topic that must be taken into account during the future of the application, now in beta and during this work, the important thing was the implementation of the application.
- Python, the language chosen for the application, adjusts very easily for the development and use of cloud services, Microsoft has full support for it, also the community behind the language is very active and helpful.
- I highlight the library for the frontend that was used, Streamlit, for its incredible easy development, which allows to have a web application deployed in a very short time.

9 References

Amazon. 2021. Cloud computing with AWS. *AWS.* [Online] 2021. https://aws.amazon.com/what-is-aws/?nc1=f_cc.

Amazon Rekognition. 2021. Amazon Rekognition pricing. *Amazon.* [Online] 01 01 2021. https://aws.amazon.com/rekognition/pricing/?nc=sn&loc=4.

Full Scale. 2019. Machine Learning in Computer Vision. *Full Scale.* [Online] 08 05 2019. https://fullscale.io/blog/machine-learning-computer-vision/.

García, Juan Antonio Piñera. 2019. Intuition behind Reinforcement Learning. *DLBT.* [Online] 04 06 2019. https://technopremium.com/blog/intuition-behind-reinforcement-learning/.

Google Cloud. 2021. Vision API. *Google Cloud.* [Online] 01 01 2021. https://cloud.google.com/vision/pricing.

Google Developers. 2020. reCAPTCHA. *Google developers.* [Online] 2020. https://developers.google.com/recaptcha/docs/v3.

Google. 2021. What is cloud computing? *Google Cloud.* [Online] 2021. https://cloud.google.com/what-is-cloud-computing.

Jiang, Richard. 2017. *Biometric Security and Privacy. Opportunities & Challenges in The Big Data Era.* Baton Rouge, Louisiana, USA : Springer International Publishing, 2017. 978-3-319-47300-0.

Kaklauskas, Arturas. 2015. *Biometric and Intelligent Decision Making Support.* s.l. : Springer International Publishing, 2015. 978-3-319-13659-2.

Kalakheti, Rachin. 2020. *Machine Learning Simplified: A Beginners' Guide To Revolution.* 2020. B087TH4N6G.

Kirk, Matthew. 2017. *Thoughtful Machine Learning with Python.* Sebastopol, CA : O'Reilly Media, Inc.,, 2017. 978-1-491-92413-6.

Korolov, Maria. 2019. What is biometrics? 10 physical and behavioral identifiers that can be used for authentication. *CSO.* [Online] 12 02 2019. https://www.csoonline.com/article/3339565/what-is-biometrics-and-why-collecting-biometric-data-is-risky.html.

McGovern, Anne. 2019. Record-breaking DNA comparisons drive fast forensics. *MIT News.* [Online] 26 06 2019. https://news.mit.edu/2019/record-breaking-dna-comparisons-drive-fast-forensics-idprism-0626.

Microsoft . 2021. What is cloud computing? *Azure.* [Online] 2021. https://azure.microsoft.com/en-us/overview/what-is-cloud-computing/.

Microsoft Azure. 2021. Cognitive Services pricing - Face API. *Microsoft.* [Online] 01 01 2021. https://azure.microsoft.com/en-us/pricing/details/cognitive-services/face-api/.

Mihajlovic, Ilija. 2019. Everything You Ever Wanted To Know About Computer Vision. *Towards Data Science.* [Online] 25 04 2019. https://towardsdatascience.com/everything-you-ever-wanted-to-know-about-computer-vision-heres-a-look-why-it-s-so-awesome-e8a58dfb641e.

Pattern Recognition. **Hancock, Edwin. 2020.** York, United Kingdom : Elsevier Ltd, 2020, Vol. 108. 0031-3203.

RAJIV CHOPRA, PhD. 2018. *CLOUD COMPUTING.* Dulles, Virginia : MERCURY LEARNING AND INFORMATION, 2018. 978-93-86286-91-8.

RaspberryPi.org. Raspberry Pi 4. *RaspberryPi.org.* [Online] https://www.raspberrypi.org/products/raspberry-pi-4-model-b/.

Statista. 2021. Global market share of cloud infrastructure services from 2017 to 2020, by vendor. *Statista.* [Online] 12 01 2021. https://www.statista.com/statistics/477277/cloud-infrastructure-services-market-share/.

Thales Group. 2020. Biometrics: definition, trends, use cases, laws and latest news. *Thales Group.* [Online] 29 12 2020. https://www.thalesgroup.com/en/markets/digital-identity-and-security/government/inspired/biometrics.

Tsai, Peter. 2018. Data Snapshot: Biometrics in the workplace commonplace, but are they secure? *Spicework.* [Online] 12 03 2018. https://community.spiceworks.com/security/articles/2952-data-snapshot-biometrics-in-the-workplace-commonplace-but-are-they-secure.

Windows Developers. 2018. Cognitive Services APIs. *Windows Blogs.* [Online] 13 02 2018. https://blogs.windows.com/windowsdeveloper/2017/02/13/cognitive-services-apis-vision/.

10 Appendix

10.1 Libraries used in Python

altair==4.1.0
appnope==0.1.2
argon2-cffi==20.1.0
astor==0.8.1
async-generator==1.10
attrs==20.3.0
azure-cognitiveservices-vision-face==0.4.1
azure-common==1.1.26
backcall==0.2.0
base58==2.1.0
bleach==3.2.3
blinker==1.4
cachetools==4.2.1
certifi==2020.12.5
cffi==1.14.4
chardet==4.0.0
click==7.1.2
cycler==0.10.0
decorator==4.4.2
defusedxml==0.6.0
entrypoints==0.3
gitdb==4.0.5
GitPython==3.1.12
idna==2.10
importlib-metadata==3.4.0
ipykernel==5.4.3
ipython==7.19.0
ipython-genutils==0.2.0
ipywidgets==7.6.3
isodate==0.6.0
jedi==0.18.0
Jinja2==2.11.2
jsonschema==3.2.0
jupyter-client==6.1.11
jupyter-core==4.7.0
jupyterlab-pygments==0.1.2
jupyterlab-widgets==1.0.0
kiwisolver==1.3.1
MarkupSafe==1.1.1
matplotlib==3.3.3
mistune==0.8.4
msrest==0.6.19
nbclient==0.5.1
nbconvert==6.0.7

nbformat==5.1.2
nest-asyncio==1.5.1
notebook==6.2.0
numpy==1.19.5
oauthlib==3.1.0
opencv-python==4.5.1.48
packaging==20.9
pandas==1.2.1
pandocfilters==1.4.3
parso==0.8.1
pexpect==4.8.0
pickleshare==0.7.5
Pillow==8.1.0
prometheus-client==0.9.0
prompt-toolkit==3.0.14
protobuf==3.14.0
ptyprocess==0.7.0
pyarrow==3.0.0
pycparser==2.20
pydeck==0.5.0
Pygments==2.7.4
pyparsing==2.4.7
pyrsistent==0.17.3
python-dateutil==2.8.1
pytz==2020.5
pyzmq==22.0.2
requests==2.25.1
requests-oauthlib==1.3.0
Send2Trash==1.5.0
six==1.15.0
smmap==3.0.5
streamlit==0.75.0
terminado==0.9.2
testpath==0.4.4
toml==0.10.2
toolz==0.11.1
tornado==6.1
traitlets==5.0.5
typing-extensions==3.7.4.3
tzlocal==2.1
urllib3==1.26.2
validators==0.18.2
wcwidth==0.2.5
webencodings==0.5.1
widgetsnbextension==3.5.1
zipp==3.4.0