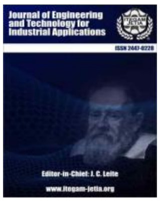
Journal of Engineering and Technology for Industrial Applications

ISSN ONLINE: 2447-0228

ITEGAM-JETIA

Manaus, v.8 n.38, p. 4-14. Nov/Dec, 2022.
DOI: https://doi.org/10.5935/jetia.v8i38.836


Journal of Engineering and Technology for Industrial Applications
ISSN 2447-0228
Editor-in-Chief: J. C. Leite
www.itegam-jetia.org

RESEARCH ARTICLE

OPEN ACCESS

ANDROID APPLICATION FOR IDENTIFICATION OF VEHICLE PLATES FOR TRAFFIC INSPECTION

Bruno Luís Litaiff Ramalho Júnior[1], Caio Augusto Barbosa Veras[2], Gabriel Melo Cavalcante[3], Thiago Caddah Rotta[4], Eliton Smith dos Santos[5] and Alexandra Amaro de Lima*[6]

[1, 2, 3, 4] Bachelor in Computer Science, Paulista University – UNIP, Manaus-Amazonas, Brazil.
[5, 6] Research in Computer Science, Paulista University – UNIP, Manaus-Amazonas, Brazil.
[5, 6] Research in Postgraduate Program in Engineering, Process Management, Systems and Environmental (PPEPMSE) - Institute of Technology and Education Galileo of the Amazon – ITEGAM, Manaus-Amazonas, Brazil.

[1] http://orcid.org/0000-0001-8988-4620, [2] http://orcid.org/0000-0002-1177-4760, [3] http://orcid.org/0000-0002-7753-6589,
[4] http://orcid.org/0000-0003-2374-7185, [5] http://orcid.org/0000-0002-8039-7532, [6] http://orcid.org/0000-0003-3918-0013

Email: bruno_carauari@hotmail.com, veras.caio12@gmail.com, gabriel__xx@hotmail.com, thiago_caddah@hotmail.com, elitonsmith@gmail.com, *xanduca@gmail.com

ARTICLE INFO

Article History
Received: November 03th, 2022
Accepted: December 28th, 2022
Published: December 31th, 2022

Keywords:
Traffic inspection,
Application development,
Solution proposal for traffic agents.

ABSTRACT

Traffic inspection is present in the daily lives of drivers. However, this inspectional process often presents itself in an inefficient and archaic way, by using paper and pen to write down the infractions committed, especially in regions lacking the latest technology. Thus, the objective of this work is to present a possible solution to this problem, with the development of a prototype application that allows the user to perform these inspections in a digital way. The methodology used includes a visit to *City Cooperation Center* (*Centro de Cooperação da Cidade* - CCC) in Manaus, Amazonas, to analyze the current scenario of the city's traffic inspection system, as well as the determination of methods to integrate the information between the application developed and PRODAM's database. At the end of the research and development, the significant potential to aid traffic agents is verified, and presents direction for future research in the area to improve what was developed in this work.

I. INTRODUCTION

Over time, new technologies are designed, developed, and tested. One of these innovations was the automobile, bringing people and places together, facilitating tasks and revolutionizing modern society. But like every invention, the automobile brought new problems and intensified existing ones, which are fought through inspections that have been intensified in recent years with the upbringing of electronic inspection that although it lacks formal coordination, the actions taken by municipal bodies when added together are of the same size as a national inspection program [1]. Today, the big urban centers in Brazil and in the world, as in our neighbors Chile and Argentina, which had their traffic codes changed respectively in 1995 and 1999 [1], count on extremely modern and powerful systems to help in the daily tasks of traffic inspectors and traffic agents, methods such as OCR (Optical Character Recognition), software for license plate recognition and even faces, as in the port city of Itapoá in Santa Catarina, that uses OCR technology to check the biometric registration of the driver and vehicle to verify if there is any cargo delivery scheduled for that day and/or time [2].

SENATRAN is the union body constituted with the purpose of representing public transport companies in the city of Manaus, responsible for enforcing traffic legislation, established by National Traffic Council (*Conselho Nacional de Trânsito - CONTRAN*). It, through its traffic agents, inspects vehicular infractions, such as stopping and/or prohibited parking, speed above the permitted limit, among others.

To carry out the inspection, traffic agents often resort to snapshots of the offending vehicle's license plate, to later register manually in the secretariat's system, so that the offender is properly treated and punished. However, this form of inspection can prove to be inefficient, as it requires a lot of repetitive and completely manual work, reducing the productivity of the service, using a slow and outdated system. How to adapt this existing system to the most practical, fast platform and using all that technology has to offer?

In the development of this work, two notebooks with different specifications are used and discussed in the materials and methods section. Several tools, described below, are also used: Figma, for designing the application screen layouts; *Android Studio*, for application development.

Code libraries, described here, are also used during the development process: Google ML KIT Text Recognition API (Google's Machine Learning Kit text recognition API), for recognizing the text on license plates; Hilt, for injecting dependencies into the application project; Data Binding, for the separation of user interface (UI) logic from business logic – that is, the code that defines the application's logical operations, not related to the graphical interface, but how it will behave upon user actions; Room Database, local database implementation abstraction library on Android; and CameraX, to handle the user's device camera.

At the end of development, the system is first tested by the developers and authors of this work, and then it is tested by the transit agents selected later to obtain feedback from them so that possible errors in the flow or operation of the application can be corrected, and, finally, an analysis of the impact of this project on the routine of transit agents is carried out.

II. THEORETICAL REFERENCE

II.1 PATTERN RECOGNITION

II.1.1 General Concepts

The most notable quality of Pattern Recognition (PR) is undoubtedly its remarkable scope when used in various fields of research [4]. Being able to act in several fields as already mentioned, with these going from the analysis of images to temperature, sound etc. Two of these areas of action will be discussed in the continuation of this work.

This area is not new, but before computers and the consequent increase in demand for them, PR did not extend to more than the field of theoretical statistics.

A complete recognition system is composed of a sensor capable of capturing the information to be analyzed (an image, sound, temperature, etc.), a machine capable of extracting, reading, and computing the information obtained by the sensor, whether this information is in numerical or symbolic and finally a model is needed to classify the acquired and analyzed data and this model in question is dependent on the data obtained.

II.1.2 Application

In terms of applications, this area is very broad as previously described, that is, even though it is unique and objectively for the analysis of captured data, these can be from different origins and for different purposes. PR is present (and in a way makes up the whole) in the analysis of satellite images, in biometric recognition (facial, voice, etc.), archival classification, among other diverse functions.

Still in terms of applicability, several PR techniques were created over time and with the growing need for data computing systems, techniques based on Artificial Intelligence (e.g., neural networks), statistical and mathematical analysis [3].

II.2 DIGITAL IMAGE PROCESSING

II.2.1 General Concepts

This field is composed of algorithms prepared for specific purposes, thus providing the user with a tool with a wide diversity of forms and processing modes. They interpret images in a mathematical way as if they were data directly related to the method of its creation [3].

In other words, this area is responsible for analyzing information from multiple dimensions captured by different sensory methods.

II.2.2 Application

Since the space race in the 20th century, this area has been gaining more and more body and strength mainly due to its applications in the beginnings of orbital flights until the advent of satellites, but we cannot forget that it is present in the solution of a variety of questions, that even not always related, need methods capable of improving visual information to make its interpretation easier for humans [5].

It is also present in areas such as medicine, where, thanks to functions such as contrast control, it allows better quality in x-ray images, ultrasounds, radiographs, among other types of imaging exams. As previously mentioned, this field also works in the meteorological area thanks to the increasingly advanced satellites.

II.3 OPTICAL CHARACTER RECOGNITION

II.3.1 General Concepts

Known by its name in English (Optical Character Recognition) it is the method that allows, as the name itself already specifies, the recognition of characters, the optical part comes from the fact that this recognition is done from the interpretation of the information contained in a file of image, the characters in this file may have been typed on a computer, written manually on a sheet of paper, printed, etc.

OCR has as its main purpose the simulation of the human capacity of reading and textual comprehension, although it is not fully capable of such a feat [6].

II.3.2 Application

OCR made it possible to create several tools that help us immensely, from performing simple day-to-day tasks such as scanning documents, reading codes through cell phone cameras, thus allowing greater automation of services, cloud text storage it, OCR, also helps in jobs that require greater detail such as analysis, reading, digitization of transcription of ancient texts by researchers and historians.

II.4 MACHINE LEARNING

The purpose of Machine Learning - or Machine Learning - is to study, design and improve mathematical models that can be trained (once or continuously) with contextualized data (from a generic environment), to infer the future and make decisions without knowledge. completeness of all influencing elements (external factors) [7]. To put it another way, machine learning consists of building programs that can adapt to the environment in which they are implemented, without having full knowledge of everything that can change their previously defined parameters. According to [8], the area of machine learning is divided into four groups, which will be discussed below.

II.4.1 Supervised Learning

In a supervised setting, the algorithms are like a student being supervised by their teachers, were, in the context of machine learning, this would be the training data for the algorithm. In this way, as the algorithm makes mistakes, it corrects itself from the data inputs it receives, like a student being corrected by his teacher [7].

In supervised algorithms, data classes are predefined, being created in a finite set, defined by a human, to be classified (the process called labeling) by the algorithm [8] (Figure 1). Within this group of machine learning algorithms, [8] comments on the existence of two main supervised models: classification and regression models.

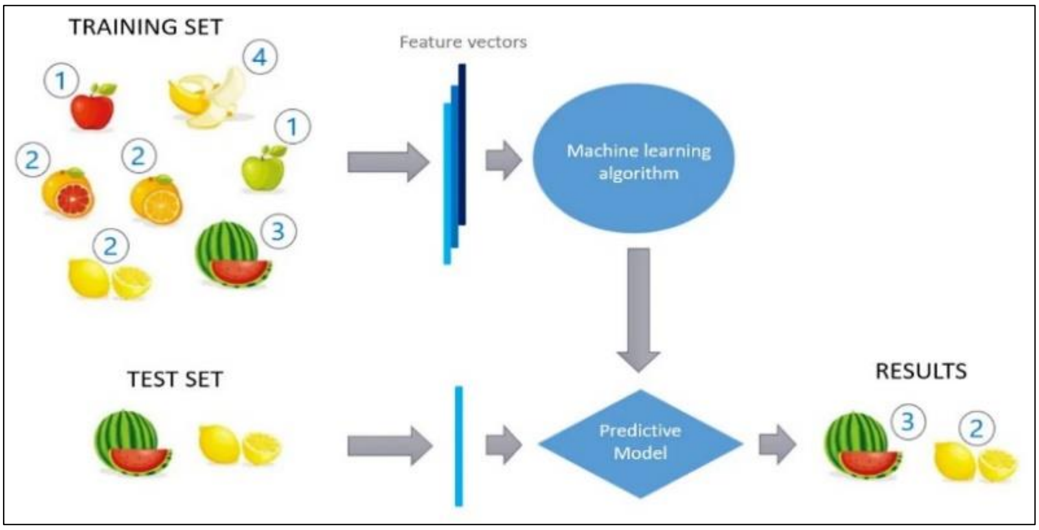


Figure 1: Supervised Learning Diagram.
Source: [9].

II.4.1.1 Classification Models

This model is a supervised learning method that consists of assigning a label to a sample based on its attributes [10].

II.4.1.2 Regression Models

Very similar to the classification model, the regression model has quantitative values as a response variable instead of qualitative ones [10]. That is, instead of predicting labels, this model aims to predict numerical continuous values [10].

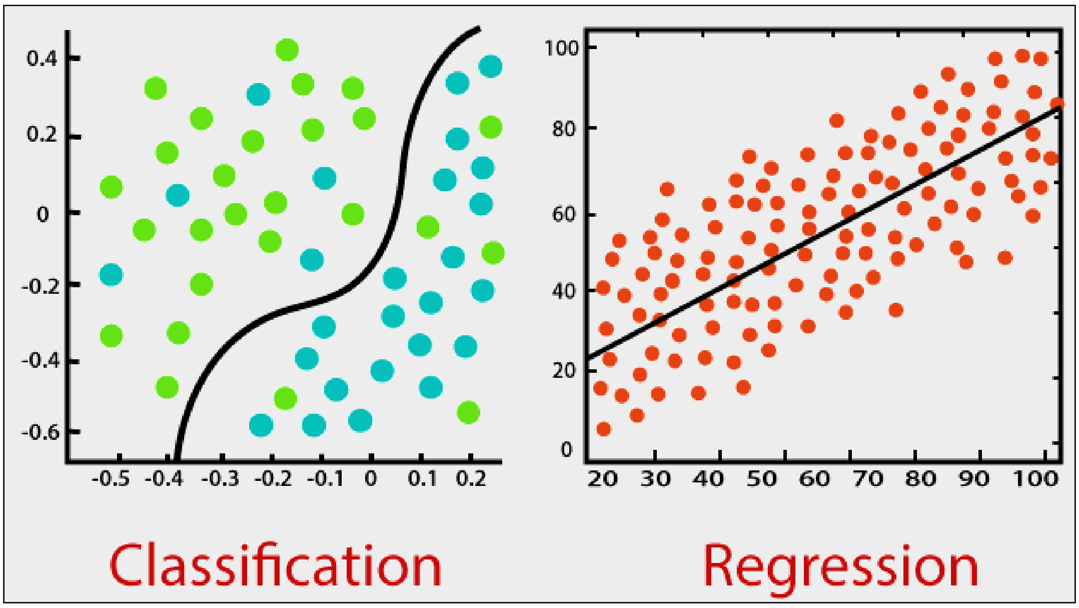


Figure 2: Classification X Regression.
Source: [11].

II.4.1.3 Use Case

A very interesting, supervised learning use case that can be noted is a study by [12] in which he used predictive analysis, which is nothing more than the application of supervised learning, to predict death cases in 5 years of life. Elderly participants in the Health Welfare and Aging (SABE) study. Another study by [12] was to predict the lifetime risk for cancer patients, with quality of up to 30 days.

II.4.2 Supervised Learning

Unsupervised learning is based on the absence of any previously defined supervisor, and thus also absent of error measures. It can be useful when the algorithm needs to learn to group (a process called clustering) a random group of elements according to their similarities [7].

Therefore, it may seem somewhat mysterious to imagine what the algorithm could learn without having much information about the inputs and expected outputs; however, its objective is not to classify, but to group by similarities [13]. Within this type of machine learning, [13] also states that there are two ways to apply unsupervised learning: clustering and dimensionality reduction, which will be dealt with in the following topics.

II.4.2.1 Clustering

It is the process of grouping similar data into different groups, that is, the partitioning of a group of data into subgroups, whose data follow whatever distance measure is defined [14]. It can also be used for the detection of anomalies in the data group, as was done in a proposal of this implementation in data from multiple views [15].

II.4.2.2 Dimensionality Reduction

Dimensionality Reduction is the process of transforming large data into a meaningful, reduced-dimensional representation. Ideally, this reduced dimension should be the same as the intrinsic dimensionality of the data, which is nothing more than the smallest dimension of parameters needed to work with the original data [16].

II.4.2.3 Use Case

An example of the use of unsupervised learning that cannot be overlooked is the video recommendation algorithm on the Youtube website. He uses the tool created by his company, Google, called Google Brain, which uses the techniques and methodologies of this learning to recommend videos that the algorithm judges (or predicts) that the user is more likely to watch [17].

II.4.3 Semi-Supervised Learning

This type of learning brings together the supervised and unsupervised learning paradigms and develops a paradigm of its own, with the aim of studying how computers and natural systems (such as human beings) learn when presented with both labeled and unlabeled data [18]. He mixes the concepts and methods defined in both paradigms to understand how this can affect the algorithm's learning outcomes, developing algorithms that take advantage of these conditions [18].

Also, according to [18], this type of learning is of great interest in machine learning and data mining, because due to its nature it can improve the solutions of the problems that supervised or unsupervised learning solves, that is, it uses the best factors of both types for a better implementation (Figure 3).

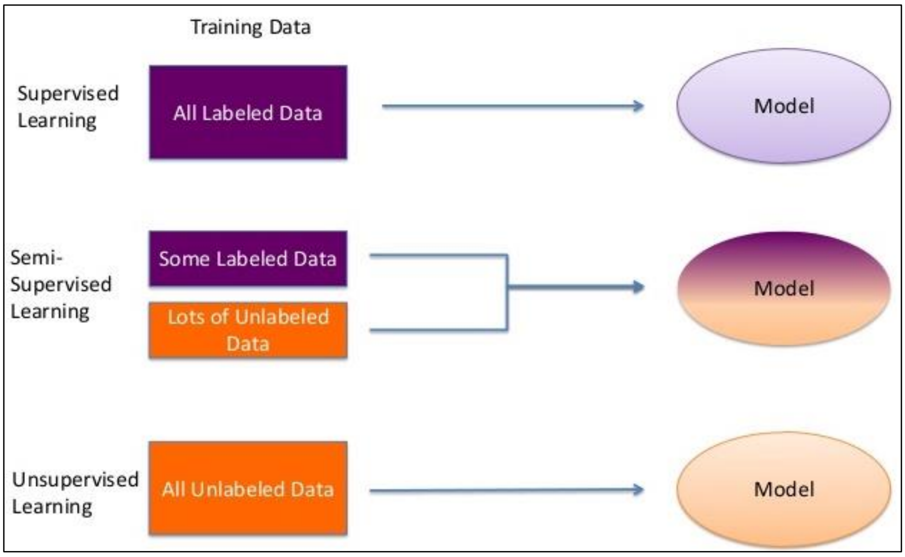


Figure 3: Diagram of the semi-supervised learning concept.
Source: [19].

II.4.3.1 Use case

An interesting use case of this type of learning is performed by [20], in which it was used to identify people in webcams.

II.4.4 Reinforcement learning

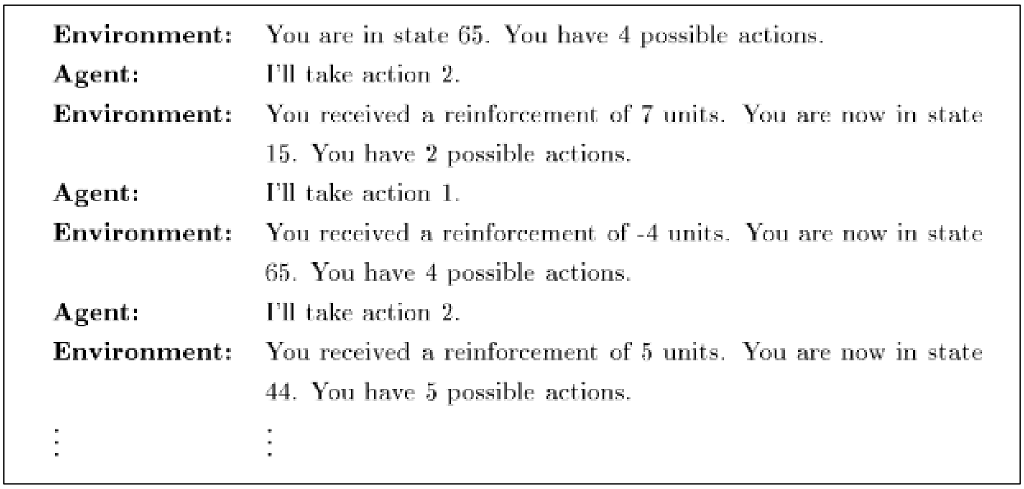
Reinforcement learning refers to the behavior of algorithms that learn through trial-and-error interactions with a dynamic environment [21]. According to [21], there are two main strategies to solve problems involving reinforcement learning. The first one is searching the behavior space to find one that performs well in the environment.

In other words, the algorithm analyzes all the behaviors that it can take when it finds itself and seeks to choose the best among them to get closer to the objective defined for it. The second consists of using statistical techniques and dynamic programming methods to estimate the usefulness of taking actions in the states of the world - that is, using statistics to determine the best action to be taken, aiming at the environment in which it is currently found.

II.4.4.1 Learning stages

The [20] says that all reinforcement learning must go through a cycle of learning stages. This cycle consists of the algorithm being presented with a state of the environment, together with the available behavior options against that current state, and then the algorithm must choose, usually by a trial-and-error system, the option that will accumulate the most points of reinforcement in the long term, which means that he will get as close as possible to the objective proposed to him. To better understand this cycle, [20] presents the following scheme:

In Figure 4, the learning cycle to which [20] refers is shown. In it, the enviroment and the algorithm (Agent) interact with each other. First, the environment tells the algorithm which state it is in, and how many and which behavior options it can adopt. The algorithm then makes the choice, and then the environment returns to the algorithm the amount of reinforcement points it received for having performed the previously chosen action. These points are what indicate to the algorithm if it has made an excellent or a bad choice, in relation to the objective to be reached. In [22] the author also discusses reinforcement learning, stating that this process basically consists of periodic state updates, with “rewards” – such as receiving more points, no points, or losing points – according to the steps taken by the algorithm. In the following, a whole reinforcement learning process will be exemplified with a maze problem, presented by [22].



Environment: You are in state 65. You have 4 possible actions.
Agent: I'll take action 2.
Environment: You received a reinforcement of 7 units. You are now in state 15. You have 2 possible actions.
Agent: I'll take action 1.
Environment: You received a reinforcement of -4 units. You are now in state 65. You have 4 possible actions.
Agent: I'll take action 2.
Environment: You received a reinforcement of 5 units. You are now in state 44. You have 5 possible actions.
⋮ ⋮

Figure 4: Reinforcement Learning Cycle.
Source: [20].

Figure 5 presents the general concept of the maze problem, which consists of making the robot (algorithm) reach the goal (gray ball). For this to happen, several phases will be covered, which will be explained below, containing related figures to facilitate understanding. On the other hand, Figure 6 shows the states of the environment, which represent the possible positions that the robot can assume. Figure 7 shows all possible movements for the robot to perform, which are moving up, right, down, and left (North, East, South and West, respectively, meaning North, East, South and West).

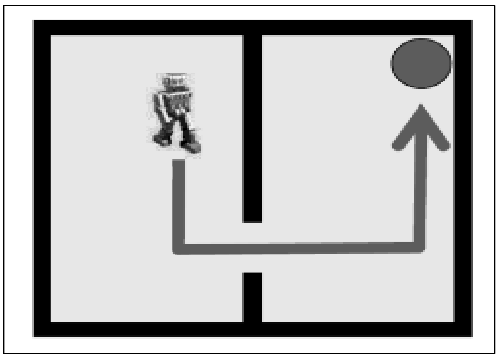


Figure 5: Maze problem.
Source: [22].

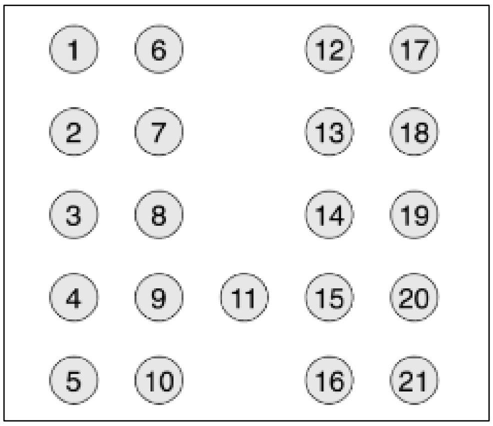


Figure 6: States of possible positions in the maze.
Source: [22].

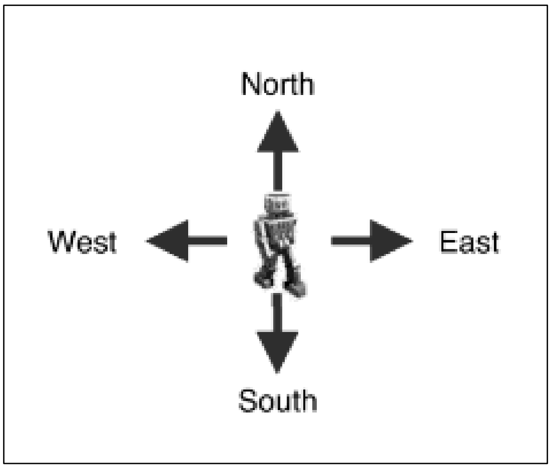


Figure 7: Possible robot movements.
Source: [22].

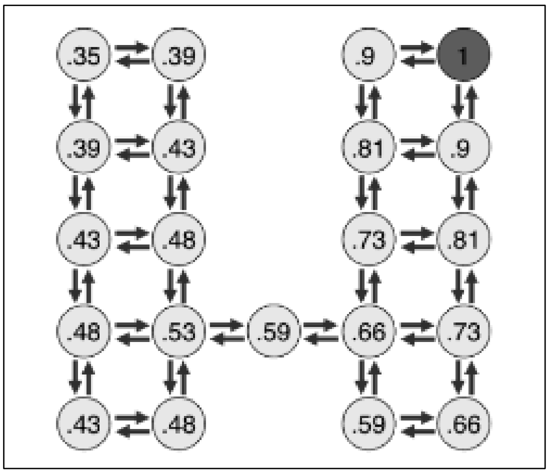


Figure 8: State transitions and rewards
Source: [22].

In Figure 8, the calculated reward values of each state are shown, which will then be used by the algorithm to decide the best action to take. One thing is important to note, that it does not matter if the path found by the algorithm is also the shortest, or the longest, but that it finds a path.

II.4.4.2 Use Case

A use case of this type of learning can be exemplified by the design of intelligent tutorials using Reinforcement Learning that assemble a dynamic and personalized study plan for each user, according to their history and performance [23].

II.4.4 Why was Android Chosen as the Target Operating System?

According to a survey by Bain & Company, 90% of smartphone users in Brazil use Android [24]. This is because Android is an open system, which allows the existence of several different brands using the system, and a variety of prices that gives accessibility to all social classes. Not only attracting users, but application developers also moved closer to Android, where for 83% of them, the fact that Android allows a highly collaborative ecosystem was the reason for the approach [24].

In the last decade, the number of users has only increased, starting at 26% and reaching an incredible 71% of users who had their first Internet experience through Android smartphones [24] (Figure 9).

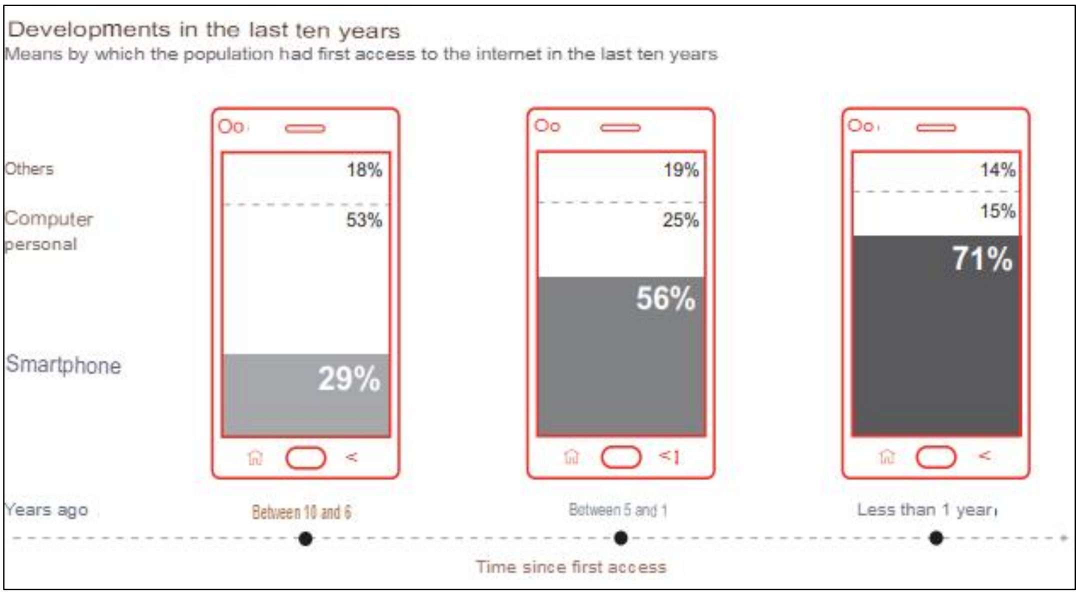


Figure 9: Internet access by Android device.
Source: [24].

With these results, we can see the great influence that Android had and still has in the lives of Brazilians. With affordable prices, modern and pleasant designs, and an easy-to-use system, the mobile operating system is the most widespread and used in Brazil, making the choice of which system to use for the development of the application of this work very easy.

III. MATERIALS AND METHODS

This section will describe the materials and methods used to implement an automotive license plate recognition system. The system will be developed specifically for the use of the security secretariat of the city hall of the city of Manaus/AM. The development of the project is important considering that the fleet is increasing in the Amazonian capital, concentrating about 81% of the fleet registered in the state. To this end, an on-site visit to *City Cooperation Center* (CCC) will be carried out first.

Opened in 2020, the CCC is an operational and intelligent center for the city and its main objective is to integrate services from the main secretariats in one place, in addition to concentrating traffic monitoring information. He has been a great ally of the city of Manaus in situations of calamity in traffic.

The CCC uses a modern data center with a dual Dell-branded server to run the system, and for the monitoring of city streets, Motorola's AVIGILON software is used, capable of storing and identifying traffic signs and human faces.

Since it is in its initial phase, the CCC is being used more as an intelligence center for the DETRAN, where the guards send their photos, and they are analyzed by the technicians who verify if the fine is correct or not. The guards use WhatsApp to send the evidence of the fine to the technicians and the technicians, after verifying the fine, send it to PRODAM, responsible for the procedures for sending the fine to the citizen.

As of December 2021, there are 855,922 vehicles out of a total of 1,054,592 vehicles registered with the State Department of Transport (DETRAN-AM). The total is 4% higher than in 2020. The IBGE estimates a population of 2.2 million people, with one car for every 2.6 inhabitants. In the continental Amazon, the proportion is one registered vehicle for every 10 residents. According to the vehicle manager of DETRAN-AM, David Fernandes, internally, there are many vehicles in circulation that are not registered due to lack of inspection [25].

III.1 MATERIALS

III.1.1 Equipment Used

For the development of the application of this work, two notebooks will be used, whose technical specifications will be discussed in Table 1.

SETTINGS	
Laptop 1	Samsung (Expert X50, Intel Core i7-8565U CPU 1.80GHz 1.99 GHz, Memória RAM 20GB, NVIDIA GeForce MX110, 1TB (HDD) TOSHIBA MQ04ABF100 & 256GB (SSD) WDC WDS240G2G0B-00EPW0, Windows 10 Home Single Language 20H2)
Laptop 2	(Acer Aspiron E 15, AMD Radeon R8 M445DX, 20GB RAM, 1TB (HDD), AMD Quad-core A10-9600p)

Table 1: Machine configuration.
Source: Authors, (2022).

III.1.2 Tools

In this section, the tools that will be used for the development of the license plate recognition application in this work will be discussed.

III.1.2.1 Figma

According to [26], Figma is a complete and easy-to-understand tool, whose simple and intuitive design allows the developer to easily build effective prototypes. The author also states that the tool is extremely practical, since it allows its use by the browser, without having to install the program on the computer.

In this way, Figma will be used to develop the layout and prototypes of the application screens, so that the application flows are planned, in addition to being possible to correct possible development bottlenecks throughout the process.

III.1.2.2 Android Studio

Android Studio is the official integrated development environment (IDE) for Android application development, built on IntelliJ IDEA, IDE for Java development [27]. According to the documentation, this tool also supports debugging, a tool that allows you to visualize the execution of specific lines of code in real time, offering a better visualization of logical problems, facilitating corrections.

In this work, this tool will be used to build the application, including the source code and visual interface of the application. In the development process, several libraries will be used to assist in the assembly of features, the main ones being discussed in the following topics.

III.1.2.3 Google ML Kit – Text Recognition

Google's library for using Machine Learning (ML) - Machine Learning - in mobile development, the Google ML Kit is the choice of every developer who needs a powerful and at the same time easy-to-implement solution [27]. According to the library's documentation, it presents several solutions that use ML, optimized for its implementation in both iOS and Android applications. All processing takes place on the device itself, making it fast and efficient, and allowing its use in offline environments. In addition, this library uses the technology that Google itself uses in its official applications, in addition to offering the developer easy-to-use APIs.

In this project, Text Recognition API (Text Recognition API) will be used. It allows recognizing texts of any Latin-based character set. According to its developers, it can also be used to automate data entry tasks like processing credit cards, receipts, and business cards.

For this work, it will be used to recognize the text of vehicle license plates, to facilitate users to send the license plate code more quickly and efficiently.

III.1.2.4 Hilt

According to [28], dependency injection is the process of efficiently building dependency graphs of other objects. According to him, dependencies are nothing more than objects required by other objects for them to work correctly. For the injection to be performed correctly, [28] states that it is necessary to decouple the construction of these dependencies from the object that requires them, passing them through the constructor of this object in question. In this way, the main object becomes more flexible for testing, since you can pass the test version of each dependency informed, making it more flexible, in addition to facilitating its usability throughout the project.

In this work, the Hilt dependency injection library, developed by Google, will be used to perform this service of providing the necessary dependencies for the project to work correctly.

III.1.2.5 Data Binding

For [29] the definition of Data Binding can be understood as being a framework that helps in the synchronization of the visual interface with data models, presenting a real-time connection of the UI and Model. In other words, this tool allows the business logic to connect directly to the visual elements that represent the data of this logic, in a performative way, simple to maintain and easy to understand.

In this project, this resource will be used in the development of the application's screens and its business logic, to make the source code simpler, smaller, and focused on what really matters, leaving the repetitive and time-consuming work for the library, streamlining the application development.

III.1.2.6 Room Database

This library, developed by Google, was built to abstract many steps of the process of building, accessing, maintaining, and changing SQLite database (Android database) [30]. According to [30], Room uses methods in interfaces that the developer can use to perform different activities with the database, to summarize the repetitive logic (boilerplate) of database configuration.

For this work, this library will be used to store the inspections carried out by the user offline, so that at another time he can synchronize them with the server (in the prototype in an emulated way), giving him comfort and security when using the application. even without a stable or even non-existent internet connection at that moment.

III.1.2.6 CameraX

According to [31], the CameraX library facilitates the implementation of camera functionality natively to Android applications, being compatible with several devices and reducing the number of repeated codes (boilerplate code).

[32] states that the library has an API that abstracts the camera's functions, which allows its use in several Android devices, using the Use-case oriented paradigm.

Its use in this work will be to capture the image of the license plate so that the area of the license plate can be cut and soon after the license plate recognition by the ML Kit library.

III.2 METHODS

In this section, the methods that will be used to achieve the specific objectives defined in this work will be presented.

III.2.1 Study of Current Conditions

To understand the status of the current system, on-site visits will be carried out to obtain all the necessary information for the development of the prototype of the application of this work.

III.2.2 Determination of Integration Methods

The integration methodology will be defined as follows: the application system will simulate the integration with the CCC system, to perform all the actions that it would perform in the de facto integrated implementation, but with this connection being emulated for testing purposes.

III.2.3 Development of the System Prototype

The application system prototype will be developed in stages, described below:

First, the layouts of the screens and the application icon will be designed. Soon after, the application will be developed, starting with the screens designed in the first stage, and ending with its logical features.

At the end of the development, the project will enter the testing stage, starting with those of the developers and authors of this work, and then, when these tests are finished, continued through on-site tests by the transit agents selected later.

III.2.4 Analysis of the Prototype's Auxiliary Potential

The analysis of the potential to help traffic agents through the complete implementation of this prototype at another time will be as follows: after the tests that were carried out by the agents of the CCC (item III.2.3) are completed, will be collected feedback on the results of these tests, so that a case study can be carried out to identify the impact that a future integrated implementation of this project would have on the routine of transit agents.

IV. RESULTS AND DISCUSSIONS

IV.1 VISIT TO THE CITY COOPERATION CENTER (CCC)

To start the activities related to the development of an automotive license plate recognition system, a technical visit was made to CCC - Manaus, with the purpose of studying the current traffic inspection systems used by the county.

At the end of the visit, it was found that there would be no possibility of developing a system of size like the one used by the CCC and, therefore, it was decided to create a prototype that simulated the interactions of the traffic agents.

IV.2 DEFINITION OF INTEGRATION METHODS

After the visit carried out at the CCC, it was possible to determine the methods that would be used to perform the integration of the prototype user with the CCC system, in an emulated way. For this, the mandatory functionalities were defined so that, in a real implementation of the proposed system, the integration of both parts could be done without problems. The prototype functionalities decided by the authors of this work were:

a) User’s login
b) Image capture
c) Recognizer for text in images
d) List of pending synchronization inspections

In this way, the integration methods were defined, and then the development of the prototype application of the solution system proposed by the authors of this work was started.

IV.3 DEVELOPMENT OF THE PROTOTYPE

IV.3.1 Application Planning

First, a name for the application was decided upon to give the project more personality. The name decided was *Agente Digital Amazonas*. The acronym of the application was determined as “ADAM”.

After deciding on the name of the prototype, some sketches were made on paper of both the screens and the flows of the application, to plan, correct and improve the initial ideas about what the prototype would be able to allow the user to do, in according to your graphical interface. Subsequently, these sketches were digitally reconstructed for better visualization.

IV.3.2 Application Programming

IV.3.2.1 Database

To persistently save both login and inspection data, a database management library called Room was used. The used version of the library was 2.4.2. The application's database was structured by two tables, called UserEntity and InspectionEntity, responsible for storing user registration information and data from each inspection performed, respectively.

IV.3.2.2 Application Screens

IV.3.2.2.1 Splash

The first screen developed for the application was Splash. Splash screens have the function of presenting the application while the initial loading of the application is carried out. For the project, this screen has the function of simulating an initial synchronization of the application with the CCC database, to emulate the use of a real case.

IV.3.2.2.2 Login

The Login screen was then developed, which corresponds to the screen in which the unregistered user will perform the registration, to have access to the application's features. In this way, this screen is responsible for user authentication (main responsibility) and for accessing the screen for adding users to the database (prototypical responsibility, this screen would not exist in the actual implementation of the project).

IV.3.2.2.3 Main Menu

It is the prototype's main screen (Figure 10). In it, the user has access to the application's features, which are:

a) View pending synchronization inspections.
b) View history of inspections carried out (and synchronized).
c) Start inspection process.

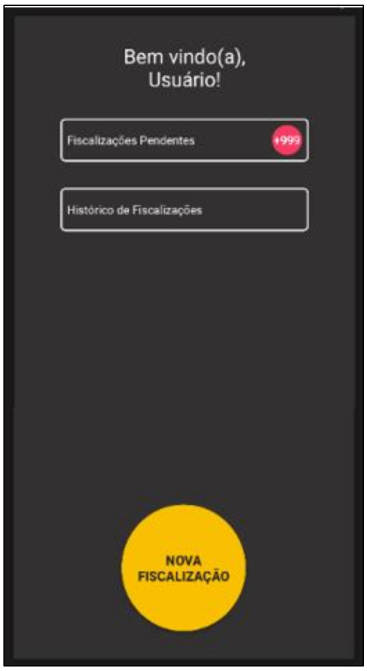


Figure 10: Main Menu Screen.
Source: Authors, (2022).

IV.3.2.2.4 Image Capture of the Vehicle License Plate

It is the screen responsible for capturing the image of the license plate to be analyzed later by the text-in-image detector (Figure 11).

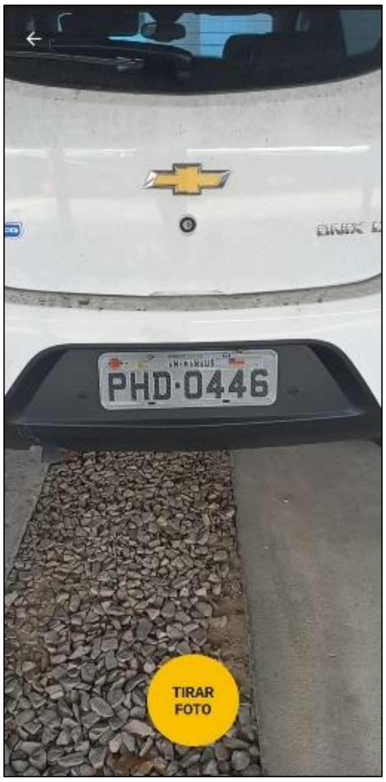


Figure 11: Car Plate Image Capture Screen.
Source: Authors, (2022).

IV.3.2.2.5 Cutout of the Vehicle License Plate Area

Screen responsible for selecting the text area of the inspected license plate, as well as identifying the text and the user's current location (Figure 12).

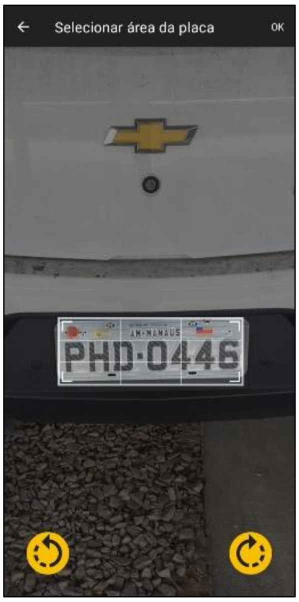


Figure 12: Vehicle License Plate Area Cutout Screen.
Source: Authors, (2022).

For text identification, the text recognition library from Google ML Kit, Text Recognizer, is used, along with cleaning and correction techniques for the results obtained from the library.

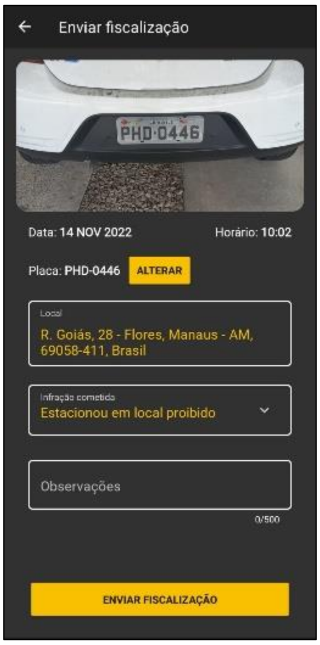
The identification of the user's current location uses the device's Geocoder together with the latitude and longitude captured from the user's GPS.

IV.3.2.2.6 List of Inspections

Screen responsible for listing inspections performed by the user, whether pending or inspection history synchronized with the system.

IV.3.2.2.7 Inspection Details

Screen with the responsibility of handling the information of each inspection (Figure 13), allowing the user to edit this data in the context of pending, or send it to the server in the context of creation of inspections.



Data: 14 NOV 2022 Horário: 10:02

Placa: PHD-0446

Local
R. Goiás, 28 - Flores, Manaus - AM, 69058-411, Brasil

Infração cometida
Estacionou em local proibido

Observações
0/500

Figure 13: Inspection Details Screen.
Source: Authors, (2022).

IV.3.3 Tests

To attest to the functionality of the application, tests of its flows were carried out, with emphasis on the process of creating new inspections in the system. To achieve the results of the tests, it was necessary for the authors of this work to carry out field tests, capturing the images of the sample vehicle plates, so that the research findings could be recorded.

The prototype tests were divided into groups, following the conditions of distance and time of day tested. For example, one of the groups refers to the test performed in an average distance between the vehicle plate and the application, in the afternoon of the day. In this way, it was possible to observe the behavior and accuracy of the application in several different situations during the day.

Figure 14 below shows some of the results obtained in capturing the license plate image during the tests performed.

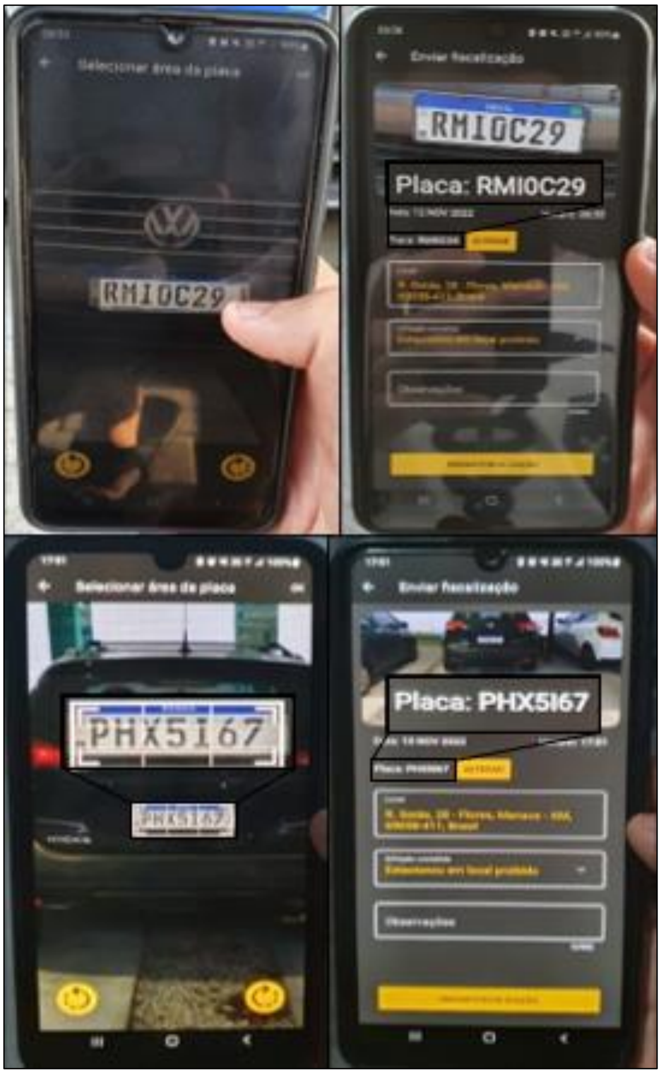


Figure 14: Application prototype tests' results.
Source: Authors, (2022).

V. CONCLUSIONS

This work intended to understand the development of a license plate recognition system in mobile devices to assist the inspection process of the traffic agents, from visits to the CCC, development of a prototype and field tests, verifying the possibility of adapting the existing system of paper inspections to the digital platform, increasing the effectiveness of traffic agents significantly, especially in regions with little technology, such as the absence of 4K cameras for detection of infractions and recognition of license plates. In future research, the prototype can be integrated with PRODAM's system, and other aspects of this system can be analyzed, to improve the functionality of the application and present traffic agents with the solution they need in the palm of their hand.

VI. AUTHOR'S CONTRIBUTION

Conceptualization: Caio Augusto Barbosa Veras and Gabriel Melo Cavalcante.
Methodology: Alexandra Amaro de Lima, Caio Augusto Barbosa Veras and Gabriel Melo Cavalcante.
Investigation: Bruno Luís Litaiff Ramalho Júnior, Caio Augusto Barbosa Veras and Gabriel Melo Cavalcante.
Discussion of results: Bruno Luís Litaiff Ramalho Júnior, Caio Augusto Barbosa Veras, Gabriel Melo Cavalcante and Thiago Caddah Rotta.
Writing – Original Draft: Gabriel Melo Cavalcante.
Writing – Review and Editing: Alexandra Amaro de Lima.
Resources: Alexandra Amaro de Lima.
Supervision: Alexandra Amaro de Lima.
Approval of the final text: Alexandra Amaro de Lima and Gabriel Melo Cavalcante.

VII. ACKNOWLEDGMENTS

The authors would like to thank the institution of City Cooperation Center (CCC) and their mentor teacher D. Alexandra Amaro de Lima for the support given during the execution of the present work.

VIII. REFERENCES

[1] A. E. R. Cannel; P. A. Gold, "Reduzindo Acidentes: O papel da fiscalização de trânsito e do treinamento de motoristas". 1º.ed. Washington: Banco Interamericano de Desenvolvimento, 2001. Available in: <https://books.google.com.br/books?hl=pt-BR&lr=&id=MwXrmVo5_K0C&oi=fnd&pg=PP7&dq=fiscaliza%C3%A7%C3%A3o+de+tr%C3%A2nsito&ots=0Uak0H06_O&sig=rhnpWgHLFjjFaTty7za5GnJb4AQ#v=onepage&q&f=false>. Access in August 22, 2022.

[2] P. S. Weber; E. M. Luz; R. J. Lunkes; D. Schnorrenberger, "Análise da Aplicação do Sistema OCR: um Estudo de Caso entre um Porto Brasileiro e um Porto Europeu". Congresso Nacional de Desempenho Portuário, 2016. Available in: <https://2016.cidesport.com.br/sites/default/files/a52193.pdf> Access in Augusto 22, 2022.

[3] J. A. de Souza, "Reconhecimento de padrões usando indexação recursiva". Tese (Doutorado em Engenharia de Produção) – Universidade Federal de Santa Catarina. Santa Catarina, p. 8-9. 1999.

[4] P. R. Meneses; T. de Almeida; A. N. Santa Rosa; E. e Sano; E. B. de Souza; G. M. Baptista; R. S. Brites. "Introdução ao Sensoriamento de Imagens por Processamento Remoto". Editora UnB, 2012.

[5] R. C. Gonzales; R. E. Woods. "Processamento de imagens digitais". Edgard Blücher, 2000.

[6] R. Mithe; S. Indalkar; N. Divekar. "Optical Character Recognition". International Journal of Recent Technology and Engineering (IJRTE), India, v. 2, nº 1, pg. 72-75, march. 2013.

[7] G. Bonaccorso. "Machine learning algorithms". Packt Publishing Ltd, 2017.

[8] V. Nasteski. "An overview of the supervised machine learning methods". Horizons. b, v. 4, p. 51-62, 2017.

[9] D. Calvo. "Supervised Learning". March 23, 2019. Available in: <https://www.diegocalvo.es/en/supervised-learning>. Access in: April 6, 2022.

[10] M. Harrison. “Machine Learning–Guia de Referência Rápida: Trabalhando com dados estruturados em Python”. Novatec Editora, 2019.

[11] JavaTPoint. "Regression vs. Classification in Machine Learning". 2022. Available in: <https://www.javatpoint.com/regression-vs-classification-in-machine-learning>. Access in: April 6, 2022.

[12] H. G. Santos. “Comparação da performance de algoritmos de machine learning para a análise preditiva em saúde pública e medicina”. Tese de Doutorado. Universidade de São Paulo. 2018.

[13] Z. Ghahramani. “Unsupervised learning”. In: Summer school on machine learning. Springer, Berlin, Heidelberg, 2003. p. 72-112.

[14] T. S. Madhulatha. “An overview on clustering methods”. arXiv preprint arXiv:1205.1117. 2012.

[15] A. A. Marcos; M. Yamada; A. Kimura; T. Iwata. “Clustering-based anomaly detection in multi-view data”. In: Proceedings of the 22nd ACM international conference on Information & Knowledge Management. p. 1545-1548. 2013.

[16] L. V. D. Maaten; E. Postma; J. V. D. Herik. “Dimensionality reduction: a comparative”. J Mach Learn Res, v. 10, n. 66-71, p. 13, 2009.

[17] M. Helms; S. V. Ault; G. Mao; J. Wang. “An overview of Google Brain and its applications”. Proceedings of the 2018 International Conference on Big Data and Education. p. 72-75. 2018.

[18] X. Zhu; A. B. Goldberg. “Introduction to semi-supervised learning”. Synthesis lectures on artificial intelligence and machine learning, v. 3, n. 1, p. 1-130, 2009.

[19] S. Jain. "Introduction to Pseudo-Labelling: A Semi-Supervised learning technique". September 21, 2017. Available in: <https://www.analyticsvidhya.com/blog/2017/09/pseudo-labelling-semi-supervised-learning-technique>. Access in: April 6, 2022

[20] M. F. Balcan; A. Blum; P. P. Choi; J. Lafferty; B. Pantano; M. R. Rwebangira; X. Zhu. “Person identification in webcam images: An application of semi-supervised learning”. ICML 2005 Workshop on Learning with Partially Classified Training Data. p. 6. 2005.

[21] L. P. Kaelbling; M. L. Littman; A. W. Moore. “Reinforcement learning: A survey”. Journal of artificial intelligence research, v. 4, p. 237-285, 1996.

[22] M. Sugiyama. “Statistical reinforcement learning: modern machine learning approaches”. CRC Press, v. 1, p. 3-7, 2015.

[23] W. Martins; U. R. Afonseca; L. E. Nalini; V. M. Gomes. “Tutoriais inteligentes baseados em aprendizado por reforço: concepção, implementação e avaliação empírica”. In: Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE). p. 550-559. 2007.

[24] L. Moura; G. Camargo. “Impacto econômico e social do Android no Brasil”. Bain & Company, [S. l.], p. 4-6, 21 set. 2020.

[25] Acrítica. “Manaus tem um veículo para cada 2,6 habitantes”. Manaus, 15 de janeiro de 2022. Available in: <https://www.acritica.com/opiniao/manaus-tem-um-veiculo-para-cada-2-6-habitantes-1.218471>. Access in: April 7, 2022

[26] J. V. C. Moura; A. B. Silva; A. A. M. Almeida; A. K. Q. Alves. “Utilização de aplicativo para abordar as características gerais das aves paraibanas”. South American Journal of Basic Education, Technical and Technological, v. 7, n. 2, p. 799-815, 2020.

[27] Android Developers. Bumblebee | 2021.1.1 Patch 2. [S.I.]. “Meet Android Studio”, 2022. Available in: <https://developer.android.com/studio/intro>. Access in: April 5, 2022.

[28] D. Prasanna. “Dependency injection: design patterns using spring and guice”. Simon and Schuster, 2009.

[29] D. Hermes; N. Mazloumi. “Data Access with SQLite and Data Binding”. Building Xamarin. Forms Mobile Apps Using XAML. Apress, Berkeley, CA. p. 347-415. 2019.

[30] R. T. Yunandar; D. Hariyanto; M. Fahmi. “Penerapan Lokal Basis Data Android Room Database (Studi Kasus: Aplikasi Ekspedisi)”. Jurnal Akrab Juara, v. 6, n. 2, p. 115-125, 2021.

[31] M. P. Gonçalves. “MLKit Text Recognition Evaluation”. Faculdade de engenharia da universidade do porto, p. 17, 2022.

[32] M. Jokel. “Implementing a TensorFlow-Slim based Android app for image classification”. Fakultat fur Informatik der Technischen Universitat Munchen. p. 23, 2020.