4COVID – Anti Covid-19 Functions for Public Transports

Tiago Silva
Instituto Superior Técnico (IST)
Lisbon, Portugal
tiago.da.silva@tecnico.ulisboa.pt

***Abstract*—In late 2019, a new virus with a high rate of spread changed the daily lives of the entire world population. Many sectors have been brutally affected by the presence of this virus, including the public transport sector. The impact in this sector is even greater because vehicles are closed public spaces, which leads people to avoid this means of mobility due to the fear of contracting the virus that can be deadly. Among the main measures to prevent the transmission of COVID-19 are the use of a mask, social distance and vehicle hygiene and ventilation, however they are not the only measures that can be implemented. This project aims to make buses safer places for people through the integration of different services. These services aim at the transmission of information that includes the real-time capacity of buses for client applications, the integration of a platform that allows obtaining customer opinions and other data about public transport services to facilitate the analysis of improvements to be made. In this way, better planning in relation to the different services will be possible, a lower percentage of occupancy on each bus and better planning by passengers on their trips.**

***Index Terms*—COVID-19; bus; passenger counting; facial recognition; face mask detection;**

I. Introduction

Currently, we are witnessing the evolution of a new pandemic that has affected the lives of all people around the globe. A new virus began to spread in China, more specifically in the region of Wuhan and, with a high degree of contagion, it spread across several countries in a few weeks. Since then, the Severe Acute Respiratory Syndrome Coronavirus 2 (SARS-CoV-2) deeply influenced people's lifestyle and mobility, with a great decrease on both air and urban circulation and an unprecedented increase in work at home. This impact clearly affected the economic and health sectors, which led governments across the world to establish measures to fight this pandemic.

Corona Virus Disease 2019 (COVID-19) represents a great challenge for all sectors of the urban mobility and public transport are no exception. This sector has presented an unprecedented decline in demand and revenue. With the rise of telecommuting, many people no longer need to move to their place of work. The use of private transport has been increasing and the use of public transport has been discouraged due to the interaction between the people in an enclosed space, although there is no evidence that a significant number of outbreaks originated or were triggered by public transport [1].

The fact that the new coronavirus is highly contagious and that many people who are infected have no symptoms substantiates a concern with regard to exposure to viruses in public transport. People find themselves confined in a closed and very limited space where the risk of contagion increases proportionally with the increase of the capacity of these vehicles. Access control does not allow the identification of sick passengers, as they may often have contracted the virus without showing symptoms. Also, the existence of multiple surfaces that are not regularly disinfected is another concern as they may be another important source of transmission of the virus. Therefore, the public transport sector has studied various forms of prevention and control that beats the pandemic so that its passengers feel safer and so that they can constitute a good alternative for urban mobility, even during the restrictions caused by the pandemic. Some of these studies involve the dematerialization of services related to public transport in order to avoid contact with potentially infected surfaces, the monitoring of stocking in order to reduce the risk of contagion, the implementation of new sanitation systems and ventilation or the implementation of systems that allow greater control by public transport operators over the safety of their passengers, namely through the analysis of certain data and events that occur during trips that may have contributed to the propagation of the viruses.

The concept of social distancing emerged as one of the greatest measures to prevent virus transmission. The World Health Organization recommends a distance of at least one meter between people, while other health-related organizations suggest physical distance of two meters. This recommended distance is for open environments, although the recommended distance for closed environments is still not known, as they have many associated variables. For public transport, physical distance is difficult to implement, because they are of closed and limited spaces, where the density of people is high and therefore the maximum capacity has been on average a third of what it was before the outbreak of the pandemic, which profoundly affects the viability of the public transport sector. Another concept that has emerged and has been shown to be one of the most effective measures is the use of face masks. Face masks can significantly reduce virus transmission in exhaled breath, particularly of asymptomatic people or with mild symptoms [2]. The filtering capacity of social masks is greater than 80% for particles smaller than 300nm and greater than 90% for particles larger than 300nm, with combinations of common tissues, such as chiffon, silk or cotton [3] [4].

Therefore, the use of masks in public spaces has been one of the main measures to fight the pandemic.

Therefore, the main objective of this project is to help the public transport sector in adapting it to the new reality, offering solutions to fight the COVID-19 pandemic. Solutions aimed at predicting future capacity, the collection of opinions and traffic information automatically through the voluntary participation of passengers and the dematerialization of bus services were solutions designed to fight the pandemic. However, due to various limitations imposed by the period of confinement due to the pandemic COVID-19, it has only been developed a solution that allows the identification of the number of passengers in a bus for a specific stop and at a specific time by counting the number of entries and exits of passengers at that time, making this information available to various modules of a system. Thus, it will be possible to alert travelers in real time and, at the same time, carry out a better management of resources by public transport companies, minimizing the impact of COVID-19 in this sector. The solution is also intended to identify the number of passengers who boarded a bus who correctly put on their face mask at that time, which helps to identify areas where greater encouragement to the use of a face mask is needed. Also, the solution will be integrated into the systems of Card4B, company responsible for software implementation on buses.

The remainder of the paper is structured as follows: in section II, topics will be addressed such as the different approaches already taken to obtain the occupation of public transport vehicles automatically, either through hardware installed in the vehicles themselves or through computer vision using cameras to obtain different types of data. The description of the passenger counting solution as well as the software and hardware used is provided in section III. The methodology used to evaluate the solution and the respective results will be done in section IV. Finally, section V describes the main findings and future work.

II. Solutions of Passenger Counting and Facial Recognition

A. Automatic Passenger Counting (APC)

A demand-driven design of public transport systems requires a full analysis of the traffic flows. To ensure a sufficient supply of public transport vehicles, it is common conduct passenger surveys repeatedly over time. Even more, in times of pandemic where physical separation between people is advised, especially in closed spaces and, consequently, the reduction of the maximum capacity allowed in transport vehicles public, this analysis has become even more urgent. Some typical issues to be resolved in this context they are obtaining the number of people inside the vehicle, the flow of passengers at each stop and analyzing the entry and exit points on the vehicle. To resolve these issues, there is an effort to automatically and continuously respond to as many questions as possible above. One of the technologies developed for this purpose is the APC technology that includes devices for installation in transit vehicles, including buses, which accurately record shipping data, thus improving accuracy and reliability the tracking of data related to the flow of passengers. APC devices can be especially useful in high-volume passenger applications and can be crucial in enabling boarding at all doors on buses to collect accurate stocking data and compare these data with the tariff revenues.

These devices provide the basis for obtaining real-time information about the agglomeration of vehicles for transit cyclists. If this information is available in real time, pilots can choose to wait for the next vehicle, if it is coming soon and less crowded. In conjunction with the efforts of the operators to adjust the service, this information can keep transit passengers safer and informed. APC data is mainly used to create, evaluate and adjust schedules and execution times, and to plan and justify route changes, but can also be used for determining the best locations to add transit points. Looking at data associated with stops from bus, those with very low capacity could be eliminated and those with very high capacity may have an additional stop added. APC also includes the option to connect video cameras and record image data, which helps in detecting fraud by comparing counter numbers with ticket machine numbers and identifying differences that can help claim higher revenue from ticketing agencies traffic and identify fraud hotspots. He can also optimize the number of passengers, using detailed data to assess the distance that the passengers actually travel, see which parts of their routes make money and which parts don't analyze the movements of passengers between the wagons and inform the requests for concessionary revenues.

To get to these answers using APC, several approaches have been developed, including light barrier sensors inside doors and switch mats, pressure sensors, onboard weighting systems, measurements of mobile signals, ticket applications or payment barriers, all with their advantages and disadvantages. But counting passengers through cameras is the most currently used approach to measure the occupation of vehicles. There are cameras that can be used in conjunction with smart counters to register a number of passengers entering and leaving at each stop or station. These video passenger counting systems can have an accuracy of 98% and with them, operators can verify that the system is counting correctly by analyzing the video showing people entering and exiting the vehicle in addition to the increasing counts. Cameras can be placed in different locations. More usually the cameras usually be placed over each door. Each of the cameras captures sequences of images and is connected to a counting unit that runs a three-dimensional software algorithm that identifies and counts individual passengers detecting entrances and exits. A built-in video server allows the users view live video images alongside people counts so they can accurately verify and configure the system via the Internet. Once configured, the counts are sent over the network to the corresponding servers or are stored locally.

B. Obtaining information using cameras

The cameras can therefore be used for various purposes and, in the public transport sector, are fundamental technologies for obtaining information that will be used in the management of many services that transport operators want to provide their customers.

People counting cameras are a widely used example to get information. Installed on buses, update the stocking information in real time, which allows a more effective control of the number of people on the bus. The people counting cameras have sensors that detect people who transit within a determined area. In addition, they contain software developed to control stocking on that area, processing the information collected and giving rise to graphs and reports that facilitate making decisions that allow to avoid overcrowding. Some models of this type of camera can identify the person's transition direction in a given area and identify whether an entry or an exit has occurred, thus giving even more tools for better stocking control while reducing resources. Yet some models contain software that shows the current capacity determined by the camera on the screen as a traffic light that informs whether the passenger will be able to enter the bus, taking into account these stocking information.

Another premise in the control of entrances to buses is the verification of the body temperature of the passengers. Usually this verification is done using IP thermometers, however in spaces with high circulation of people such as buses, the measurement turns out to be very time-consuming. Soon, in these spaces it is necessary to implement a body temperature measurement system that uses thermography, such as thermal cameras. These cameras are capable of converting the radiation from infrared emitted by human beings in a graphic image measuring accurately, in real time, the body temperature of people, thus being able to detect cases of fever instantly. Furthermore, they incorporate artificial intelligence for facial detection, which helps to filter out other heat emitters that the camera can capture. All thermal camera solutions contain a software for managing and monitoring the temperatures of people who, after detecting a temperature above a preset value, activate an audible or visual alarm.

However, cameras are specially used to capture images that can be used for facial detection and recognition. Facial detection and recognition technologies are widely used for different tasks in different areas. In all these tasks, face detection aims to identify faces from the people through captured images and videos and facial recognition allows to identify the individual's biological data. There are other biometric systems that allow you to identify people using fingerprints or iris, but these systems have the disadvantage of involving a process where it is necessary contact with the person. Facial analysis systems can capture images from a distance without there is any physical contact with the person to be identified, which in times of pandemic is seen as a great advantage over other systems. Easy image storage for facial analysis can be seen as another advantage of these systems [6]. Currently there are Application Programming Interfaces (API's) that do this facial analysis, as the more used are Amazon Web Services (AWS) face recognition API and Microsoft Azure Facial Recognition API.

AWS face recognition API analyzes images containing faces, either for facial detection or for face comparison, bringing together demographics such as people's age and gender. For face detection, this API determines the presence, location, scale and orientation of any face present in an image or in a video, regardless of attributes such as gender and age. For facial recognition, it receives an image and makes a prediction about the correspondence with other faces identified in others images present in a database. For both face detection and face recognition, it provides an estimate of the confidence level of the prediction in the form of a confidence score. The region with the largest confidence score contains a higher probability of detecting a face or detecting faces correspondents. Through these confidence scores, users must consider the threshold of the confidence score provided by the API when designing their applications and thus decisions according to the output of this [7].

The Microsoft Azure Facial Recognition API is an API developed by Microsoft that detects and compares human faces, organizes images into groups based on similarities, identifies people previously tagged in images, running locally or in the cloud. Plus you can check the probability that two faces belong to the same person, returning a confidence score about the probability that the two faces belong to one person. The face cognitive service provides several facial analysis functions including detection, verification, identification and grouping of faces. In face detection, it returns the coordinates of the bounding box that "finds" the face, being able to extract a series of attributes related to the face. At face checking, it checks if two faces belong to the same person or if one face belongs to a certain identified person. When identifying faces, this cognitive service creates a group of people by analyzing faces within a database and, after creating and training that database, identifies whether a face detected in a new image belongs to that group of people. In face grouping, it divides a set of unknown faces into multiple groups of people based on similarity aspects, such as the faces being the same age, or the same gender or until similar emotions are identified [8].

III. Passenger counting and face mask detection solution

As referred before, the proposed solution will be integrated into Card4B's integrated mobility solutions. Fig. 1 shows a functional architecture of the system initially designed after the integration of the solution into the system already implemented by Card4B. However, within the scope of this work, it was only possible to implement the system in the area marked in blue, which includes a solution in the Sensors module that provides information about the passenger count and mask detection, also providing the images captured by the cameras at the time of the passenger flow.

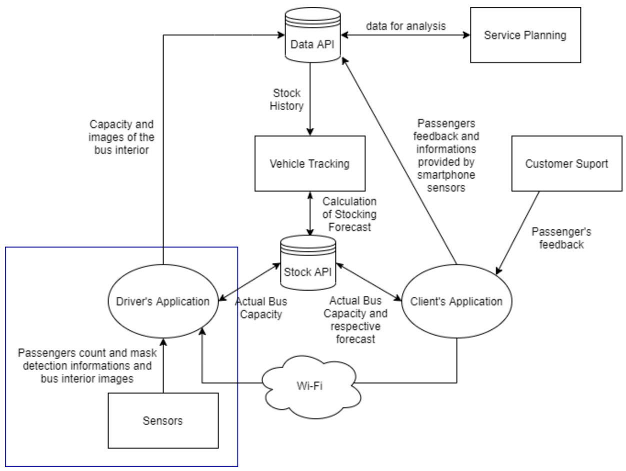

Fig. 1. Overall solution architecture

The solution presented consists of counting passengers at each stop using cameras. Two passenger counting cameras installed at the entrance and exit doors will be used. Each one contains sensors that detect and count people who pass through a specific area and the information collected by these will be processed later. The camera acquired to carry out the passenger count was the SUNMI Face Sense Camera, a camera widely used for monitoring stores. This camera will be used to monitor the entrances and exits of passengers on buses where it is necessary to analyze the interface present in the SUNMI App. In this way, the Sensors module is able to remove the number of entries and the number of exits that the camera was able to detect and with them calculate the current capacity on the bus, subtracting the number of exits from the number of entries. To access the results of this count, it is necessary to use an API that SUNMI provides for this purpose, called OpenAPI. Due to the unavailability of using OpenAPI due to problems found in OpenAPI and due to the delay in solving these problems, the counting process had to be remodeled, starting to be done through the analysis of the captured images by the camera, which were manually placed in a storage solution. Due to this restriction, it is not necessary to capture images at the bus doors using a specialized counting camera, it is just enough to install a camera that captures several images at certain time intervals during the process of entering and leaving passengers at a stop.

This solution involves two states: after opening of doors and after closing of doors. In each of these states the system will have different functions. In the first state, images will be captured by cameras installed on the bus doors during the flow of passengers. In the second state, the analysis of the images captured in the first state will take place, including the count of passengers during this flow and verification of the use of a mask. Verification of mask use will only be made on images captured by the camera installed on the front door.

After the first state, it is necessary to send the images captured by the cameras to an image storage server. To store the images, Azure Blob Storage was used, which is a Microsoft's object storage solution for the cloud. To send the images, there must be an exchange of messages between Azure Blob Storage and the Sensors module. When the bus doors close, after passengers enter and exit it, its server asks the Azure Blob Storage server for permission to send the images captured by the cameras. Then the Azure Blob Storage server asks for the bus identifier, the trip identifier and the stop identifier respectively in order to validate the request. Offering valid values, the Sensors module will provide the images to the Azure Blob Storage server which will organize the images and place them in the correct folders.

With the images available in Azure Blob Storage, it is possible to calculate the number of people who entered and left the bus. For this, it is necessary to have a local meter in order to accumulate the number of passengers who are currently on the bus. The objective is to identify passengers in the images present in the Azure Blob Storage, checking if these passengers have already been identified, and also to identify if this passenger is included in the number of entries or exits. Thus, it is necessary to isolate a folder with images that served as a reference for identifying passengers, comparing them with the faces identified in the images. As software for detecting and verifying faces, the Microsoft Azure Facial Recognition API was used. After detecting a face in an image, that face will be compared with the faces of passengers that have been detected before the flow of passengers at that stop. If the comparison returns a match, it means that the passenger was already on the bus before the image was captured, otherwise the detected face will be placed in the isolated folder.

Fig. 2 illustrates the process for counting people and for mask detection. The Sensors module starts by searching all the images captured by the camera installed in the entrance door that are located in the folder "*imagesIn*" of the Azure Blob Storage. For each image, the Microsoft Azure Facial Recognition API detects the faces present in the image. Each of the detected faces is compared with the faces located in the "*persons-db*" folder (that matches with the faces of the people who are already on the bus), in order to verify if that person was already on the bus before the entry process. This phase is very important for the final solution, as it avoids cases where the same person contributed to more than one increment in the counting process and allows the isolation of people who were already on board before the passenger entry process at a particular stop. Thus, if the API cannot identify the face captured in the database, it will create a notification to increase the passenger counter that will then be sent to the Sensors module. Furthermore, that face will then be placed in the "*persons-db*" folder, as the person with that face will now belong to the group of passengers on board the bus. If the API cannot identify the face captured in the database, it proceeds to the mask detection process, checking if the person whose face was detected is using a mask and if it is completely covering his mouth and nose. During this analysis, if the API fails to identify the correct use of the mask, the Sensors module will increment the counter for the variable "Defaulters" in order to identify a passenger who did not comply with the rule regarding the placement of the face mask when entering the bus. Then the API analyzes the next image and the process continues in the same way, until there are no more images to analyze in the "*imagesIn*" folder.

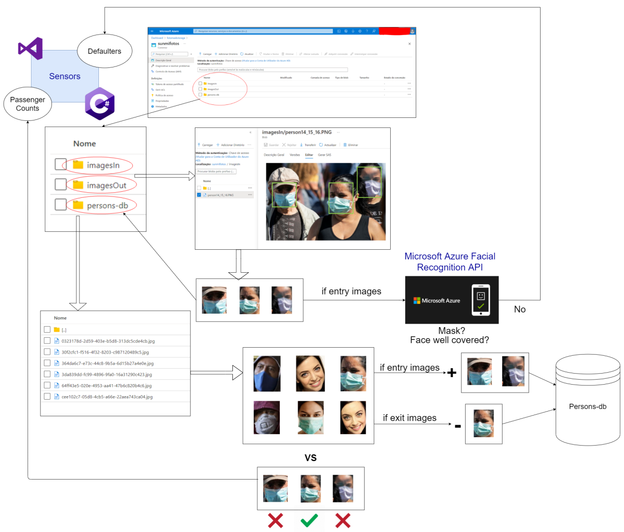

Fig. 2. Process for passenger counting and mask detection

After analyzing all images captured by the camera installed at the entrance door, all images captured at the exit door will be analyzed. The process for counting passengers who departed at a particular stop is very similar to the process for counting passengers who entered. The local server starts by looking for images present in the folder "*imagesOut*" present in the Azure Blob Storage. For each of these images, the Microsoft Azure Facial Recognition API detects the faces present in the image. Then, each of the detected faces is compared with the faces present in the "*persons-db*" folder, in order to verify if that person was already present on the bus before the exit process. If the API is able to identify the face captured in the database, it will create a notification to decrement the passenger counter, which will then be sent to the Sensors module. Furthermore, that face will then be removed from the "*persons-db*" folder, as the person with that face has left the bus and therefore no longer belongs to the group of passengers on board the bus. Then the API analyzes the next image and the process continues in the same way, until there are no more images to analyze in the "*imagesOut*" folder.

Finally, the Sensors module will send the data to the Driver's Application. The communication between these two modules is done through two web services installed in each one of the modules. The web service installed in the Sensors module will receive the identifiers whose data the Driver's Application intends to receive and will send the respective data and vice versa, constituting a bidirectional communication between modules. The data intended by the Driver's Application includes the number of passengers currently on the bus, the number of people who can still get on the bus, the number of entries and exits during the flow of passengers at that stop, a timestamp to temporally locate that flow, the trip identifier and the number of passengers whose mask was not detected. If the Sensors module identifies that there was no detection of the correct use of the mask on a passenger or even if the local counter contains a number equal to or greater than a maximum capacity limit, its web service will send a notification of the identified problem to web service from the Driver's Application.

IV. Methodology and Results

This section aims to evaluate the performance of the developed system. It will first test the position of the cameras in order to minimize errors in the counting of passengers made by the SUNMI Face Sense Camera. Then, several images of the same person will be analyzed in order to test the Microsoft Azure Facial Recognition technology in relation to the identification of a person in an image having as a comparison term another image already present in the system database. Finally, it will test the system for counting passengers and detecting face masks, using a test scenario in which the entry and exit of passengers along a route with thirteen stops was simulated, using images that were added to Azure Blob Storage.

A. Passenger Counting with SUNMI Face Sense Camera

Although the data obtained by SUNMI Face Sense Camera cannot be captured due to the problem on the OpenAPI, it is important to understand at which angles the cameras capture images that are analyzed with the least possible conditions, in order to get the best results. Therefore, it is concluded that it is very useful to analyze these results even without the use of these cameras in the implemented solution. In order to test the passenger count, made by cameras in real time, the SUNMI Face Sense Camera was mounted at two different angles, representing two scenarios: one representing the diagonal movement of the passenger and the other representing the vertical movement of the passenger, from the perspective of the camera. The purpose of these tests is to simulate entrances and exits on a bus, testing the count of movements that, crossing a given virtual line, symbolize entry or exit on a bus. For this the image will have to be adjusted first, focusing the person's face in a face recognition box and delineating the limits of the door. This adjustment is represented in Fig. 3, corresponding to scenarios A and B. If the detected move is performed from left to right, that move will symbolize an entry. If on the other hand the detected movement is performed from right to left of the line, that movement will symbolize an exit.

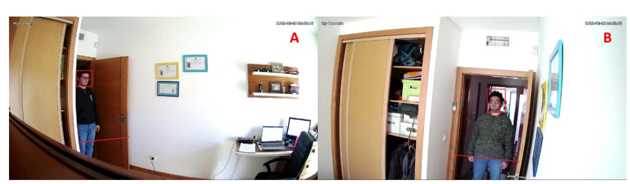

Fig. 3. Scenarios captured by SUNMI Face Sense Camera

To evaluate the results, a success rate was calculated for each scenario that consists of the relationship between the passages counted by the camera and the actual passages, as shown in (1).

$$Rate\ of\ Success = \frac{Passages\ Counted}{Actual\ Passages} \tag{1}$$

For each of the scenarios, seven tests will be performed: only with one person, with two people at the same time, with a person wearing a mask, with a person wearing a hat, with a person wearing sunglasses, in the dark with one person and in the dark with two people at the same time. The results of the seven tests performed for scenarios A and B are shown in the tables presented in Fig.4.

Tests/Scenarios	Scenario A						Scenario B					
	Entries			Exits			Entries			Exits		
	Real	Detected	Success rate	Real	Detected	Success rate	Real	Detected	Success rate	Real	Detected	Success rate
1 person without accessories	20	20	100%	20	15	75%	20	16	80%	20	14	70%
2 people	40	38	95%	40	34	85%	40	30	75%	40	32	80%
Wearing mask	20	17	85%	20	17	85%	20	16	80%	20	12	60%
Wearing a hat	20	20	100%	20	16	80%	20	14	70%	20	10	50%
Wearing sunglasses	20	20	100%	20	11	55%	20	13	65%	20	17	85%
In the dark: 1 person	20	20	100%	20	18	90%	20	17	85%	20	15	75%
In the dark: 2 people	40	36	90%	40	35	87.5%	40	31	77.5%	40	27	67.5%

Fig. 4. Results from passenger counting tests with SUNMI Face Sense Camera

After analyzing the results from scenario A, it was noticed that the camera has a greater difficulty in counting entries of people wearing a mask, which is justified by the fact that it occupies a greater percentage of the person's face compared to a hat or sunglasses, which can make it difficult to detect person's face by the face detection mechanism present in the camera. As for exits, it was possible to conclude that the camera has slightly lower exit success rates compared to entries, which can be explained by the impossibility to see the passenger exiting beyond the edges of the door, which may cause confusion on the counting sensors. In addition, it is also concluded that, unlike the case of the entries, the camera has more difficulty in detecting people who are wearing a hat or especially sunglasses than people who are using a mask, because, in the process of exiting, the camera captures a smaller percentage of the person's face, capturing the upper part of the face more easily than the lower part.

After analyzing the results from scenario B, it was possible to understand that the camera has more difficulties in detecting entrances than in the previous scenario, which can be explained by the light coming from the outside and that it is backlight when the camera is pointed directly at the door, as opposed to another scenario. It was also noticed that in this scenario, the camera has more difficulties in counting people who wear a hat or sunglasses compared to people who wear a mask, which can be explained by the fact that they occupy the eye area, the main indicator for facial recognition, especially when the camera is pointed in front of the person. As for the exit scenarios, there were not so many differences in relation to the entry scenarios compared to scenario A, which can be explained by the fact that, in this case, the camera was able to capture the passenger after the exit, due to the extension of the image in relation to the delineated virtual line, contrary to what happens in scenario A. It was also noted a greater difficulty for the camera to detect exits when the passenger uses the hat and mask in relation to the use of sunglasses, as sunglasses do not occupy a percentage of head as high as the previous two. Note that the difference in results may also have been derived from the different degrees of camera calibration for the two scenarios.

These tests allow us to get conclusions about the placement of the camera in bus entry and exit scenarios in different situations. For a better analysis, the values of the entry-exit combinations in all scenarios were calculated through the average of the success rate of the entries and exits. Through this last analysis, the results of an entry-exit scenario were averaged, in order to obtain the overall success rate in the defined entry-exit set. The results are shown in the table in Fig. 5. It is possible to understand that the best approach is to place the entry and exit cameras in a scenario similar to Scenario A, as it was the situation where there was a higher overall success rate with around 88%.

Entry/Exit	Scenario A	Scenario B
Scenario A	87.67857142857143%	82.67857142857143%
Scenario B	76.19897959183673%	71.55612244897959%

Fig. 5. Overall average success rate on each entry-exit set

B. Test for the validation of faces in the solution

To test the validation of faces in the solution, the images represented in Fig.6 will be used. The objective is to confirm that the detected face belongs to the same person in relation to the person whose face was used as a comparison term. To do this, one of the ten images was placed on the Azure Blob Storage database in the "*persons-db*" folder and it was compared with the remaining images, one by one, placed in the "*imagesIn*" folder. The table shown in Fig.7, indicates the results after performing this test, with the lines showing the image that was defined as a comparison term (called **verified images**) and the columns representing the remaining images that were used to verify that the identified face is the same (referred to as **validable images**). The red spaces indicate that the system's response was that the faces do not belong to the same person, that is the trust variable was less than 0.5, while the green spaces indicate the highest trust value for each one of the scanned images.

With these results, it can be concluded that image 6 is the one with the least confidence in the system, since the system did not consider that the face identified in this image is the same as the face identified in all other images. A possible justification is the fact that image 6 presents the person with his face back to the camera, and thus not being so visible. Images 1 and 5 also presented very negative results, because image 1 is out of focus, making it more difficult for the system to identify characteristic features of the face, and image 5 presents a perspective where only a small percentage of both areas of analysis is visible.

The results also demonstrate that the images that generate the highest values in the confidence variables are those that

Fig. 6. Images used for comparison: 1- frontal position, out of focus; 2- front position, with infrared light; 3- front position, with mask; 4- front position, with sunglasses; 5- lateral position, with mask, without glasses; 6- on the back, with mask; 7- frontal position, with sunglasses and mask; 8- frontal position, without glasses, with hat; 9- lateral position, with mask, nose outside; 10- front position, with hat

	1	2	3	4	5	6	7	8	9	10
1						0.92068				
2			0.62567	0.72642				0.806		0.79667
3		0.7044		0.66567	0.72642		0.64839	0.60684	0.61556	0.71209
4		0.76116	0.68423				0.61838	0.79935		0.84897
5									0.5501	
6										
7			0.66504	0.59448						
8		0.80902	0.65819	0.78419	0.53461					0.93757
9	0.75005		0.73657	0.65426	0.57344		0.55003			
10		0.8245	0.70984	0.86041	0.5413			0.93816	0.56938	

Fig. 7. Confidence values for different image comparisons. Lines: Verified Images; Columns: Validable Images. In red: no match; In green: highest match for each scanned image

have image 3 verified. Image 3 shows the person using the mask correctly and with glasses on, which means that the system is able to adapt well the different variables identified on the face of the validable image, whether it has sunglasses and a hat, or wearing a mask, since the system can always identify the similarity on the face in at least one of the analysis zones on the face. With this test it was also possible to conclude that the glasses do not have a great influence on the analysis of the eye area, as the confidence values in this case were very positive.

Also for image 10, the system presents good values, validating on six occasions, excluding image 7. This factor is due to the fact that image 7 has sunglasses and mask, which causes that both the eye and mouth area are not very similar compared to image 10 where these utensils are not used. For image 4, the system showed negative results for images 5 and 9, due to the difference that exists in the eyes area, justifiable by the use of sunglasses by the person in image 4, and in the mouth area, justifiable by the use of mask both in images 5 and 9. For image 8, the system did not validate images 7 and 9, which can be justified by the same reasons used for image 10. For image 9, the system showed negative results for the images 2, 8 and 10, because in the image 9 the area of the person's mouth is covered by a mask unlike images 2, 8 and 10.

For image 2, the system did not validate images 5, 7 and 9 due to the fact that the person was wearing sunglasses and a mask in image 7 and due to the different framing in images 5 and 9. For image 7, the system did not validate images 2, 5, 8, 9 and 10 because it shows the person wearing sunglasses and a mask, which does not happen in any other highlighted image, except the images 5 and 9 which present the face in a different perspective. Image 3 shows the person with the mask on and image 4 shows the person with the sunglasses on and both in the same perspective in relation to image 7, which means at least one of the analysis zones that the system considers is similar between the two images.

With these results, it can be concluded that the correspondence is made, in general, if any of the validable image analysis zones is similar to the corresponding zone in the verified image. It is still possible to conclude that the correspondence between faces is easier to obtain, when both the verified image and the validable image present the person in a frontal position to the camera. It can also be concluded that the use of infrared to capture photographs in dark environments does not contribute to the non-validation of images by the system. On the other hand, it is necessary that the captured images have a good resolution and not appear blurry, as this contributes a lot to the ineffectiveness of the system in this aspect. Another conclusion to be drawn from these tests is the recommendation to use a mask when leaving the bus. During entry, the correct use of a mask is required by the passenger, so it will be an image of the passenger with a mask that will be used as a verified image in the face comparison process. Therefore, if the passenger also uses the mask at the exit, the validable image will contain the nose and mouth area similar to the verified image corresponding to that passenger, which will simplify the process of comparing faces and, consequently, the process of counting passengers by the system.

C. Test Scenario

To simulate the functioning of the system in real time, a test scenario was carried out and includes thirteen stops where images were analyzed for the entry and exit of passengers in a given bus. The maximum capacity considered was eight passengers. Stop identifiers were numbered and sorted according to the order in which the bus arrived at the stops. Due to the constraints caused by the COVID-19 pandemic, it was not possible to perform tests in a physical environment. Due to OpenApi connectivity problems (platform that provides images captured by the SUNMI Face Sense Camera), it was not possible to capture moving images and simulate an input or output scenario of a passenger on a bus. It was therefore necessary to use images taken from the Internet to simulate several cases that can create difficulties for the passenger counting system. So, a few images were used for each stop in order to facilitate the analysis of results. In a real environment more images would have to be captured to avoid passengers that are not captured. The images introduced in the previous section will be used, with the aim of simulating the identification of the same person using different images. These images, in addition to the remaining images that were taken from the Internet, are stored locally, representing a total of twenty-five people. For the exit of passengers, some print-screens of images that were used for the entrance, were used in order to represent the same people who entered at the time of exit, due to the difficulty in providing different images of the same person. This factor should positively affect passenger validation, increasing the system's effectiveness.

The calculation of the current capacity is done using the formula in (2).

$$Capacity\ Current = On\ board + Entries - Exits \quad (2)$$

The variable "*On board*" consists of the number of identified people that are on the bus before the flow occurs, which corresponds to the number of files located in the folder "*persons-db*" on that moment. The variable "*Entries*" consists of the number of people who were identified in the process of entering who were not yet on board the bus. Thus, whenever a person is identified in this process and is not detected in the database, the variable "*Entries*" is incremented, and it is reset to zero when this process ends. Therefore, a person detected at the entrance contributes to the calculation of the current capacity only if the answer to the question "On board?" is negative. The variable "*Exits*" consists of the number of people who were identified in the process of leaving who were already on board. When a person is identified in this process and their face is detected in the database, this variable is incremented. Therefore, a person detected on departure contributes to the calculation of the current capacity, only if the answer to the question "On board?" is positive. As for the variable "*Entries*", the variable "*Exits*" is reset to zero when the passenger exit process is finished.

The system also aims to determine the number of people who are not respecting the rule of correct placement of the mask when entering the bus. For that, the system uses the Microsoft Azure Facial Recognitial technology whose functionalities include the detection of the face mask in a person's face and if it is well placed. With this information, it is possible to count the number of people who are not using the mask correctly, through a variable "*Defaulters*" which is incremented whenever the person detected at the entry process is present in the database and whose response to the question "Well covered?" is negative.

Given the scenario presented, some characteristics of the people detected in the images that will challenge the system should be highlighted, as the rest do not have restrictions that could change the practical result in relation to the theoretical:

1) The image showing person 3 shows that he is wearing a transparent mask around his mouth.
2) In the image that encompasses people 6 to 13, there are people who appear in the background, and their faces are partially covered by the people who appear in the foreground.
3) Person 14 is facing away from the camera.
4) Persons 24 and 25 are twins.

The system obtained some results different from what was theoretically correct. The table represented in Fig. 8 presents these results by comparing them with the theoretically predicted results. In the theoretical results column, people who were not subsequently detected by the system are shown in green, and in the results column, the errors resulting from the execution of the system are shown in red. The same table also shows the time in seconds that the system took to process the images and obtain the results at each stop. The criteria used for checking and validating faces in this scenario are the same as those used in the previous section where several images of the same person were compared.

Establishing a sequential analysis, the first system error happened during the mask detection process of person 3 at stop 1. Person 3 is using a face mask that covers the nose and mouth, corresponding to the correct position in the placement of the mask. However this area is transparent and therefore visible. When analyzing the image, the system can detect the mouth, considering that they are not covered by the mask. Thus, the system detects that the mask was not correctly placed, contrary to reality, thus increasing the number of defaulters.

The second error of the system also occurred at stop 1 during the facial recognition process for the exits when the system cannot recognize the person 1 who left the bus, so it does not increase the value of the exits. This is because the system recognized that the passenger was not on the bus, while he actually was. One justification for this result is that, as shown in the previous section, images 1 and 5 of person 1 obtain a confidence value lower than 0.5. This error affects not only the number of departures, but also the current capacity considering as two passengers on board, while in reality there was only one.

The third error occurred at stop 3, where the system recognized the persons 7 and 10 who were getting on the bus but not yet on board. The image captured in this stop contains some people who are covered by other people who appear on a higher plane. These people were identified as person 7 and 10 in the table and their face is quite covered, which makes it difficult to detect and recognize their faces. If for detection there were no problems, for recognition there were and the difficulty of facing the face for easier analysis made the system recognize the face in another image that does not correspond to the same person. In this way, the count of the number of entries was affected and consequently the capacity after the flow of passengers at stop 3. This error caused greater consequences in the results since at stop 4, the system did not consider the bus as full as the current capacity was still below the maximum capacity of eight passengers, while in reality it was not. In this way, one more passenger was allowed to enter the bus, again affecting the current capacity value.

The fourth error occurred in the entry process at stop 5, where the system did not detect person 14, again affecting the counting process. The failure to detect the person 14 was due to the fact that the person 14 is turning back to the camera. As the detection of people is done by detecting their faces, then it is impossible for the system to detect that person, so it can be concluded again that the images taken by the cameras

StopId	Theoretical Results	Practical Results	Runtime (seconds)
0	**People on board:** Empty **Entry**: 2 people • Person 1 -> Mask? Yes; Well covered? Yes; On board? No • Person 2 -> Mask? Yes; Well covered? No; On board? No **Exit**: 0 people **Current stocking**: 0+2-0= 2 people **Defaulters:** 1 person	**People on board:** Empty **Entry**: 2 people • Person 1 -> Mask? Yes; Well covered? Yes; On board? No • Person 2 -> Mask? Yes; Well covered? No; On board? No **Exit**: 0 people **Current stocking**: 0+2-0= 2 people **Defaulters:** 1 person	6
1	**People on board:** 1, 2 **Entry**: 1 person • Person 3 -> Mask? Yes; Well covered? Yes; On board? No **Exit**: 2 people • Person 1 -> On board? Yes • Person 2 -> On board? Yes **Current stocking**: 2+1–2=1 person **Defaulters:** 0 people	**People on board:** 1 **Entry**: 1 person • Person 3 -> Mask? Yes; Well covered? No; On board? No **Exit**: 2 people • Person 1 -> On board? No • Person 2 -> On board? Yes **Current stocking**: 1+1–1=2 person **Defaulters:** 1 person	5
2	**People on board:** 3 **Entry**: 2 people • Person 4 -> Mask? Yes; Well covered? Yes; On board? No • Person 5 -> Mask? Yes; Well covered? Yes; On board? No **Exit**: 0 people • Person 1 -> On board? No **Current stocking**: 1+2-0=3 people **Defaulters:** 0 people	**People on board:** 1, 3 **Entry**: 2 people • Person 4 -> Mask? Yes; Well covered? Yes; On board? No • Person 5 -> Mask? Yes; Well covered? Yes; On board? No **Exit**: 1 person • Person 1 -> On board? Yes **Current stocking**: 2+2-1=3 people **Defaulters:** 0 people	60
3	**People on board:** 3, 4, 5 **Entry:** 8 people • Person 6 -> Mask? Yes; Well covered? Yes; On board? No • Person 7 -> Mask? Yes; Well covered? Yes; On board? No • Person 8 -> Mask? Yes; Well covered? Yes; On board? No • Person 9 -> Mask? Yes; Well covered? Yes; On board? No • Person 10 -> Mask? Yes; Well covered? Yes; On board? No • Person 11 -> Mask? Yes; Well covered? Yes; On board? No • Person 12 -> Mask? Yes; Well covered? Yes; On board? No • Person 13 -> Mask? Yes; Well covered? Yes; On board? No **Exit:** 2 people • Person 3 -> On board? Yes • Person 4 -> On board? Yes **Current stocking**: 3 + 8 – 2 = 9 people **Defaulters:** 0 people	**People on board:** 3, 4, 5 **Entry:** 8 people • Person 6 -> Mask? Yes; Well covered? Yes; On board? No • Person 7 -> Mask? Yes; Well covered? Yes; On board? Yes • Person 8 -> Mask? Yes; Well covered? Yes; On board? No • Person 9 -> Mask? Yes; Well covered? Yes; On board? No • Person 10 -> Mask? Yes; Well covered? Yes; On board? Yes • Person 11 -> Mask? Yes; Well covered? Yes; On board? No • Person 12 -> Mask? Yes; Well covered? Yes; On board? No • Person 13 -> Mask? Yes; Well covered? Yes; On board? No **Exit:** 2 people • Person 3 -> On board? Yes • Person 4 -> On board? Yes **Current stocking**: 3 + 6 – 2 = 7 people **Defaulters:** 0 people	118
4	**People on board:** 5, 6, 7, 8, 9, 10, 11, 12, 13 **Entry**: Autocarro cheio (isFull = true) **Exit:** 2 people • Person 8 -> On board? Yes • Person 9 -> On board? Yes **Current stocking**: 9 + 0 – 2 = 7 people **Defaulters:** 0 people	**People on board:** 5, 6, 8, 9, 11, 12, 13 **Entry**: 1 person • Person 1 -> Mask? No; Well covered? No; On board? No **Exit:** 2 people • Person 8 -> On board? Yes • Person 9 -> On board? Yes **Current stocking**: 7 + 1 – 2 = 6 people **Defaulters:** 1 person	64
5	**People on board:** 5, 6, 7, 10, 11, 12, 13 **Entry:** 5 people • Person 14 -> Mask? Yes; Well covered? Yes; On board? No • Person 15 -> Mask? Yes; Well covered? Yes; On board? No • Person 16 -> Mask? Yes; Well covered? Yes; On board? No • Person 17 -> Mask? Yes; Well covered? Yes; On board? No • Person 18-> Mask? Yes; Well covered? Yes; On board? No **Exit:** 3 people • Person 4 ->; On board? No • Person 5 -> On board? Yes • Person 13 -> On board? Yes **Current stocking**: 7 + 5 – 2 = 10 people **Defaulters:** 0 people	**People on board:** 1, 5, 6, 11, 12, 13 **Entry:** 4 people • Person 15 -> Mask? Yes; Well covered? Yes; On board? No • Person 16 -> Mask? Yes; Well covered? Yes; On board? No • Person 17 -> Mask? Yes; Well covered? Yes; On board? No • Person 18-> Mask? Yes; Well covered? Yes; On board? No **Exit:** 3 people • Person 4 ->; On board? No • Person 5 -> On board? Yes • Person 13 -> On board? Yes **Current stocking**: 6 + 4 – 2 = 8 people **Defaulters:** 0 people	118
6	**People on board:** 6, 7, 10, 11, 12, 14, 15, 16, 17, 18 **Entry:** Autocarro cheio (isFull = true) **Exit:** 4 people • Person 6 -> On board? Yes • Person 16 -> On board? Yes • Person 17 -> On board? Yes • Person 18 -> On board? Yes **Current stocking**: 10 + 0 – 4 = 6 people **Defaulters:** 0 people	**People on board:** 1, 6, 11, 12, 15, 16, 17, 18 **Entry:** Autocarro cheio (isFull = true) **Exit:** 4 people • Person 6 -> On board? Yes • Person 16 -> On board? Yes • Person 17 -> On board? Yes • Person 18 -> On board? Yes **Current stocking**: 8 + 0 – 4 = 4 people **Defaulters:** 0 people	60
7	**People on board:** 7, 10, 11, 12, 14, 15 **Entry:** 1 person • Person 1 -> Mask? Yes; Well covered? Yes; On board? No **Exit:** 4 people • Person 11 -> On board? Yes • Person 12 -> On board? Yes • Person 14 -> On board? Yes • Person 3 -> On board? No **Current stocking**: 6 + 1 – 3 = 4 people **Defaulters:** 0 people	**People on board:** 1, 11, 12, 15 **Entry:** 1 person • Person 1 -> Mask? Yes; Well covered? Yes; On board? Yes **Exit:** 3 people • Person 11 -> On board? Yes • Person 12 -> On board? Yes • Person 3 -> On board? No **Current stocking**: 4 + 0 – 2 = 2 people **Defaulters:** 0 people	58
8	**People on board:** 1, 7, 10, 15 **Entry**: 2 people • Person 19 -> Mask? Yes; Well covered? No; On board? No • Person 20 -> Mask? Yes; Well covered? Yes; On board? No **Exit:** 1 person • Person 10 -> On board? Yes **Current stocking**: 4 + 2 – 1 = 5 people **Defaulters:** 1 person	**People on board:** 1, 15 **Entry**: 2 people • Person 19 -> Mask? Yes; Well covered? No; On board? No • Person 20 -> Mask? Yes; Well covered? Yes; On board? No **Exit:** 1 person • Person 10 -> On board? Yes **Current stocking**: 2 + 2 – 1 = 3 people **Defaulters:** 1 person	13
9	**People on board:** 1, 7, 15, 19, 20 **Entry**: 3 people • Person 21 -> Mask? Yes; Well covered? Yes; On board? No • Person 22 -> Mask? Yes; Well covered? Yes; On board? No • Person 23 -> Mask? No; Well covered? No; On board? No **Exit:** 2 people • Person 7 -> On board? Yes • Person 19 -> On board? Yes **Current stocking**: 5 + 3 – 2 = 6 people **Defaulters:** 1 person	**People on board:** 1, 19, 20 **Entry**: 3 people • Person 21 -> Mask? Yes; Well covered? Yes; On board? No • Person 22 -> Mask? Yes; Well covered? Yes; On board? No • Person 23 -> Mask? No; Well covered? No; On board? No **Exit:** 2 people • Person 7 -> On board? No • Person 19 -> On board? Yes **Current stocking**: 3 + 3 – 1 = 5 people **Defaulters:** 1 person	61
10	**People on board:** 1, 15, 20, 21, 22, 23 **Entry**: 1 person • Person 24 -> Mask? No; Well covered? No; On board? No **Exit:** 2 people • Person 1 -> On board? Yes • Person 21 -> On board? Yes **Current stocking**: 6 + 1 – 2 = 5 people **Defaulters:** 1 person	**People on board:** 1, 20, 21, 22, 23 **Entry**: 1 person • Person 24 -> Mask? No; Well covered? No; On board? No **Exit:** 1 person • Person 1 -> On board? No • Person 21 -> On board? Yes **Current stocking**: 5 + 1 – 1 = 5 people **Defaulters:** 1 person	109
11	**People on board:** 15, 20, 22, 23, 24 **Entry**: 2 people • Person 24 -> Mask? No; Well covered? No; On board? Yes • Person 25 -> Mask? No; Well covered? No; On board? No **Exit:** 3 people • Person 15 -> On board? Yes • Person 24 -> On board? Yes • Person 25 -> On board? Yes **Current stocking**: 5 + 1 – 3 = 3 people **Defaulters:** 2 people	**People on board:** 1, 20, 22, 23, 24 **Entry**: 2 people • Person 24 -> Mask? No; Well covered? No; On board? Yes • Person 25 -> Mask? No; Well covered? No; On board? Yes **Exit:** 3 people • Person 15 -> On board? No • Person 24 -> On board? Yes • Person 25 -> On board? No **Current stocking**: 5 + 0 – 1 = 4 people **Defaulters:** 2 people	63
12	**People on board:** 20, 22, 23 **Entry**: 0 people **Exit:** 4 people • Person 20 -> On board? Yes • Person 21 -> On board? No • Person 22 -> On board? Yes • Person 23 -> On board? Yes **Current stocking**: 3 + 0 – 3 = 0 people **Defaulters:** 0 people	**People on board:** 1, 20, 22, 23 **Entry**: 0 people **Exit:** 4 people • Person 20 -> On board? Yes • Person 21 -> On board? No • Person 22 -> On board? Yes • Person 23 -> On board? Yes **Current stocking**: 4 + 0 – 3 = 1 person **Defaulters:** 0 people	10

Fig. 8. Comparison between the results desired for the execution of the system and the results obtained in the proposed type scenario

have to capture people facing forward. This error caused a change in the current stocking count but did not cause as many consequences as the previous error as the current stocking continued to be lower than the maximum stocking.

The fifth error occurred in the exit process at stop 10, namely in the recognition process for the image corresponding to person 1, since the system was not able to recognize it within its database. This error was expected by viewing the table of the previous section where it is shown that the comparison between images 2 and 7 have confidence values lower than 0.5, so the system considers that the faces detected do not belong to the same person. From this analysis, it can be concluded that this error caused the existence of passengers after the outflow at the last stop, since the person identified as still on board is person 1 who should have been removed from the database in the validation process corresponding to the exits at stop 10.

The sixth and final error occurred at stop 11 during the person 25 recognition process. In the test scenario, person 25 enters and exits at the same stop, so an image of him was captured during both streams. However, in both situations, the system recognized person 25 as person 24. This fact is explained by the similarity of the faces of people 24 and 25, as they are twins. In this way, the system obtained confidence values greater than 0.5 and thus determined that the faces belonged to the same person.

In conclusion, the presented system is a practical way of counting passengers in real time, while being able to detect the correct use of a face mask when passengers enter buses. The system has a high degree of reliability, being able to recognize a person in several specific contexts. However, it is an unreliable system in contexts where the person's face is barely visible in the image capture. Therefore, if the cameras that capture the images during the flow of passengers are well positioned to capture as many faces as possible and as frontally as possible, and if the captured photographs are of sufficient quality for face verification, these problems can be avoided. The system also had some problems in rarer case studies, such as the use of transparent masks or the entry of twins, however they will not be very significant for the intended purpose.

Another fact that should be taken into account is the time interval considered between one photograph taken and the next so that the faces of all people who entered and left the bus are captured, minimizing the system's running time. This requires real-time tests that were not possible to carry out in the scope of this work. On the other hand, the running time of the system can be a little high compared to the intended one. This test scenario use very short samples and yet at some stops the system took about two minutes from the start to the end of the one stop counting process. Taking into account that some trips between stops can take approximately this time interval, the counting information would hardly be available to passengers with some time in advance of the arrival of the bus to the stop they want. Furthermore, passenger flows on buses can double or triple compared to the flows shown in the previous scenario, which means the system runtime will also increase.

Therefore, this system is not recommended for routes with a short time interval between stops and with a large flow of people. The use of this system is more suitable for bus trips with few stops and large time intervals between stops, namely intercity trips or for buses rented by companies in order to displace few people.

V. Conclusions and Future Work

A. Conclusions

The main objective of this paper is to help Card4B in adapting to a new reality created by the COVID-19 pandemic, offering solutions that ensure the needs of passengers. For this purpose, a passenger counting solution was developed in order to control the capacity of buses in real time. The solution consists of a system capable of detecting faces through images captured by cameras during the flow of passengers, and in turn, counting the number of different faces detected in a set of images, providing data about the number of entries, exits and current capacity of the bus after the flow of passengers at a particular stop. For this, SUNMI Face Sense cameras, specialized in counting people, would be used. However it was not possible due to the unavailability of the API of these cameras to supply the data obtained by the cameras to the system. Thus, the detection and counting of passengers were carried out by the system with the aid of facial recognition technologies. In addition to the control of bus capacity, the aim was also to obtain the number of passengers who correctly used the face mask when entering buses, a measure that has so far been mandatory by the standards of health authorities in some countries. To this end, the system also includes detecting the correct use of the face mask and the respective count for each stop. Much of the system's development resulted in the integration of Microsoft's facial recognition API. Due to the conditions imposed by COVID-19 it was not possible to integrate the system with the system belonging to the Card4B company, nor to carry out tests in a real environment nor as real images, that is, in the daily life of a bus trip. However, the system was tested in a simulated scenario, using images locally stored.

First, tests were carried out in order to determine the best position of the cameras on the bus, comparing two positions. From the results obtained, it can be concluded that the best position among the tested ones is the diagonal position for both the input and output processes, obtaining an average success rate of about 88%. Then, the system was tested in a simulated travel scenario. Based on the results obtained, the system is capable of counting passengers in normal situations, managing to detect and validate faces consistently. However, the system is unreliable in contexts where the person's face is not visible in the image capture, which requires that the cameras that capture the images during the flow of passengers are well positioned in order to detect the faces more effectively and as clear as possible. The photographs should have sufficient quality for checking faces too, avoiding blurry as much as possible. Regarding the execution time, it is concluded that the system is not recommended for routes with a short time interval between stops and that contain a large flow of people, being more recommendable for bus trips with few stops and large time intervals between stops.

B. Future Work

The main aspect to be finalized in the system is its integration into the Card4B system, namely the inclusion and implementation of the web services used for the exchange of data between the Sensors module and the Driver Application, and the testing of the system in a real environment and in a real time. Only after this test is it possible to conclude the overall performance of the system in terms of effectiveness and efficiency. Another aspect to improve is the inclusion of the SUNMI Face Sense Camera in the counting process, after the correct functioning of the API included in it to supply the data obtained to the developed system. In this way, it will increase the efficiency of the solution, since the counting of passengers does not require the detection and validation of faces by the system, considerably reducing the system's execution time.

On the long term, the aspects to be improved will include the implementation of the other features referred to in this article, namely the prediction of the capacity of a given bus for future stops on its route using an algorithm that combines the capacity history of previous days with the current capacity of the bus, the analysis of statistical management indicators and contagion probabilities, based on occupations in buses, stops, lines, sections or areas where the volume of contacts is greater or the dematerialization of transport services, through the integration of an interface more digital between the various actors in a bus journey.

Acknowledgment

The author would like to thank Card4B Systems for the involvement and the commitment throughout the project.

References

[1] Janette Sadik-Khan and Seth Solomonow. (2020, June) Fear of Public Transit Got Ahead of theEvidence. (accessed: 14-11-2020). [Online]. Available: https://www.theatlantic.com/ideas/archive/2020/06/fear-transit-bad-cities/612979/

[2] K. A. Prather, C. C. Wang, and R. T. Schooley, "Reducing transmission of sars-cov-2,"Science, vol.368, no. 6498, pp. 1422–1424, June 2020.

[3] A. Konda, A. Prakash, G. A. Moss, M. Schmoldt, G. D. Grant, and S. Guha, "Aerosol filtration efficiency of common fabrics used in respiratory cloth masks,"ACS nano, vol. 14, no. 5, pp. 6339–6347, May 2020.

[4] A. Tirachini and O. Cats, "Covid-19 and public transportation: Current assessment, prospects, and research needs,"Journal of Public Transportation, vol. 22, no. 1, p. 1, June 2020.

[5] C. Rodier and E. Issac, "Transit performance measures in california," April 2016.

[6] O. Y. Nikolaievskyi, O. Skliarenko, and A. Sidorchuk, "Analysis and comparison of face detection apis," , no. 4, pp. 39–45, 2019.

[7] A. F. Recognition. Visão geral da detecção de faces e comparação de faces. (accessed: 09-05-2021). [Online]. Available: https://docs.aws.amazon.com/ptbr/rekognition/latest/dg/face-feature-differences.htmll

[8] M. A. F. Recognition. O que é o serviço Azure Face? (accessed: 10-05-2021). [Online]. Available: https://docs.microsoft.com/pt-pt/azure/cognitive-services/face/overview