Journal of Autonomous Intelligence (2021) Volume 4 Issue 2.
doi: 10.32629/jai.v4i2.500

Original Research Article

Application of Artificial Intelligence in Monitoring the Use of Protective Masks

Alexandre Pereira Junior[1], Thiago Pedro Donadon Homem[1], Fabio Oliveira Teixeir[1]

[1] Federal Institute of Education, Science and Technology of São Paulo, São Paulo 3550308, Brazil

ABSTRACT

In the context of current epidemic diseases, this study developed a web application, which can monitor the use of protective masks in public environments. Using the Flask framework in Python language, the application has a control panel to help visualize the obtained data. In the detection process, Haar Cascade algorithm is used to classify faces with and without protective masks. Therefore, the web applications are lightweight, allowing the detection and storage of images captured in the cloud and thte possibility of further data analysis. The classifier presents precision, reversal and f-score of 63%, 93% and 75%, respectively. Although the accuracy is satisfactory, new experiments will be carried out to explore new computer vision technologies, such as the use of deep learning.

Keywords: *Computer application, COVID-19, Facial detection, Haar Cascade, Artificial intelligence, Prevention*

ARTICLE INFO

Received: Aug 18, 2021
Accepted: Oct 10, 2021
Available online: Oct 17, 2021

***CORRESPONDING AUTHOR**

Alexandre Pereira Junior
a. Junior@aluno.ifsp.edu.br;
Thiago Pedro Donadon Homem
thiagohomem@ifsp.edu.br

CITATION

Junior AP, Homem TPD, Teixeir FO. Application of Artificial Intelligence in Monitoring the Use of Protective Masks. Journal of Autonomous Intelligence 2021; 4(2): 40-51. doi: 10.32629/jai.v4i2.500

1. Introduction

At the end of 2019, Wuhan recorded the first undetected disease case in the world. Affected people presented cough, sneezing, fever, and the worst case is severe pneumonia, leading to acute respiratory failure[1]. The patient's clinical condition deteriorated rapidly and the intensive care unit was required to make efforts to restore health. In just a few days, Wuhan has become the main epicenter of the new virus, and the world continues to strive to control the epidemic, such as building hospitals in a record time and strictly implementing social distance measures. An example of these measures is , which aims to limit personal contact with infected individuals, which is necessary for infectious diseases without vaccines or treatment. Therefore, Wuhan was isolated to curb the spread of the virus.

The new virus was named SARS-CoV-2. It has been identified as part of the coronavirus family, the pathogen of COVID-19, a new disease. The spread of this disease around the world has led to a public health emergency of international significance, which is described by WHO as an epidemic. Due to the high transmission rate of the virus, isolation measures need to be taken; It occurs from one person to another by touching, dripping saliva, sneezing, coughing, phlegm, contaminated objects or surfaces. According to the World Health Organization (WHO), basic personal hygiene care can reduce the transmission of the virus, such as hand hygiene with alcohol and gel, and the use of personal protective equipment, such as surgical masks (N95) or handmade masks.

Studies have shown that masks can block 95.15% to 99.98% of aerosols[2].

In addition, the mortality rate of COVID-19 is about 3.4%, which is 9.6% and 34.3% respectively compared with other coronaviruses such as SARS and SRM[3]. In some cases, the clinical development of the virus requires the use of ventilator supported intensive care units (ICU). Due to the speed of disease transmission, some countries face difficulties in meeting the demand for these devices and avoiding as many deaths as possible. Therefore, social isolation and personal protection measures have been essential to reduce the number of infections and avoid overburdening the health system. Countries that postponed such measures, such as Italy and Spain, faced the problem of hospital overcrowding, resulting in thousands of cases and deaths due to the lack of beds, and finally have to decide who would receive appropriate care and who would not.

In Brazil, the first action taken to curb the spread of the disease was implemented at the airports to prevent the entry of people with specific symptoms of the disease, such as cough, fever, runny nose, sore throat, respiratory distress, loss of smell (anosmia), taste disturbance (ageustia), gastrointestinal diseases (nausea, vomiting, diarrhoea), tiredness (asthenia), decreased appetite (hyperoxia) and dyspnoea (shortness of breath). However, the entry of the virus is inevitable, because at some stage of the disease, the infected person is asymptomatic.

Due to population density and human trafficking, the state of São Paulo has become the epiccenter in the country. The city has the busiest airport in South America, providing air bridges for various flights. After confirming the first batch of COVID-19 cases, the state announced quarantine. Educational institutions and non-core activities have been prevented from continuing to operate and are not expected to reopen.

The largest city in the state of São Paulo, according to Decree 59396[4], everyone must wear masks in public places, shops and public transport, otherwise they will be fined. Masks have become an essential element for São Paulo residents to move around the city in an emergency. Because even with the efforts of the government of São Paulo, the population segregation rate is very low; in the period from April to May 2020, this proportion is between 46% and 59%[5], ideally 70%.

Data to measure quarantine compliance are collected through a system that maps population mobility indices, known as the intelligent monitoring system of Sao Paulo[5]. The system is established through public-private partnership and involves national mobile operators. Although SIMI-SP is effective in identifying aggregates and measuring isolation rates, it does not monitor public behavior when crossing public places, such as the use of protective masks.

In order to complement the actions of SIMI-SP, it is necessary to develop a monitoring system to determine whether individuals use personal protective equipment in public places where people circulate. Action of this nature is necessary until an effective drug or vaccine COVID-19 is found.

The use of artificial intelligence (AI) can monitor the behavior of people in transit during a pandemic. According to Pontes[6], AI is a computing field aimed at stimulating problem-solving and decision-making capabilities. According to Backes and Junior[7], their sub field "computer vision is a research field that attempts to transfer amazing visual ability to machines", because computers interpret data differently from humans. Artificial vision focuses on [...] Capture and enhance the image (for example, eliminate noise, increase contrast, etc.), separate the region or object of interest from the scene, extract various information such as shape, color and texture from the analyzed image, and finally associate the image with other previous views.

An improved monitoring method of artificial intelligence is facial recognition. It is used in many fields, from the most aesthetic — for example, to improve photos using the face as the main element in applications such as Instagram. — to other areas such as agriculture and police skills[8]. Face recognition can be used to monitor a large number of people at the same time, highlight their faces, and use algorithms to convert these images into data that can be understood by computer systems.

1.1 Ask questions

Due to the high infectivity of SARS-CoV-2, the use of protective masks is mandatory in several countries as an effective and low-cost alternative to curb the spread of the virus, because it protects the individuals who use it and the people around them. According to the study, even in the case of reduced cases, wearing masks and other measures, such as social distance, may continue until 2022[9]. Therefore, autonomous social monitoring solutions are needed to promote the recovery of economic and social activities by ensuring that measures continue to be taken during and after the most critical period of the pandemic.

Thus, the main purpose of this study is to develop a composite solution to monitor the use of protective masks in public and private environments from the images taken by the camera by using AI and computer vision technology. To that end, the following specific objectives are proposed: 1) Using artificial vision and AI algorithm to build a face detection system; 2) Develop a database to store the obtained data; 3) create a dashboard to monitor the system data.

2. Feasibility study

This section describes initiatives and studies related to the proposed project. A computational solution is proposed to minimize the pan demia impact of COVID-19.

2.1 Related work

Even without face detection technology, it is worth monitoring SIMI-SP[10] monitoring system currently used by the state government of São Paulo. The system produces a map of urban agglomerations and can monitor the rate of social isolation in urban centers and places most conducive to the spread of new cases. It was created by a public-private partnership involving mobile operators. However, this solution cannot determine whether a person is using personal protective equipment, such as protective masks recommended by who and mandatory in some cities.

In September 2020, the "face recognition" descriptor was searched in the virtual science article Library of Web of Science, and 10325 times in the past five years. In this result, 5737 articles were published in scientific journals, 4371 in scientific conferences, and 217 in other ways. Only 116 articles (about 1.1%) were written by Brazilian authors. Although they account for only a small part of the research, some Brazilian institutions have discussed the use of face recognition in their projects.

One of the publications discusses the gender classification of people in the image, identifying whether a person is male or female in real time in an uncontrolled environment through their physical characteristics, and factors that may impair the accuracy of the image (e.g. Site brightness). In order to solve this problem, the author simulated human understanding of gender recognition and established a neural network to recognize these patterns[11].

Another publication of Brazilian scientists is the Deepfake study[12]. This technology uses AI to create false videos, which may lead to false positives in face recognition systems. By using convolutional neural network (CNN), the security system can benefit from the solution proposed in this paper.

Among the selected articles, a Brazilian study also compared traditional machine learning and deep learning methods[13]. According to this article, the model created from the trend database performs better in face recognition task.

2.2 Reason

In view of the above, in the case of a high-risk pandemic, it is necessary to monitor the use of masks by the public in the public environment to ensure the effectiveness of measures to prevent covid-19 infection. This article describes the technology used to build a solution for autonomously monitoring the use of personal protective masks by people in an uncontrolled environment.

3. Architecture solutions

The following describes the architecture of the solution proposed in this project and the relationship between the computing components used.

3.1 Algorithm classifier

The classifier used in this application is based on the technology proposed by Viola and Jones[13] and the *Haar Cascade* method[14], which has been specially used for face, human and other aspects so far. However, the classification process requires image preprocessing.

The image in the computer system is a three-dimensional matrix composed of color, width and height. In the image processing step, they define their default height and width so that they all have the same pixel size. In the process of image processing, some noise can still be removed from the image obtained in the capture process, such as smoothing and highlight filtering. In this work, we decided to omit the dimension of color and convert the image to gray. In this way, the classifier will execute the function in a two-dimensional array.

Haar Cascade method[15] includes three basic steps: 1) track the image and find the Haar features similar to the bus object; 2) use boosting classifier to select the most relevant features; 3) cascade classifiers to improve the final result.

Therefore, in the first stage, it scans the image by rotating small particles from left to right and from top to bottom to find Hal's characteristics[16]. **Figure 1** illustrates the type of manual core used to detect Haar Cascade: image (a) reference core to identify edge features; image (b) reference texture is used to identify the straight line feature, and the image (c) reference texture is used as four rectangles to identify the diagonal edge.

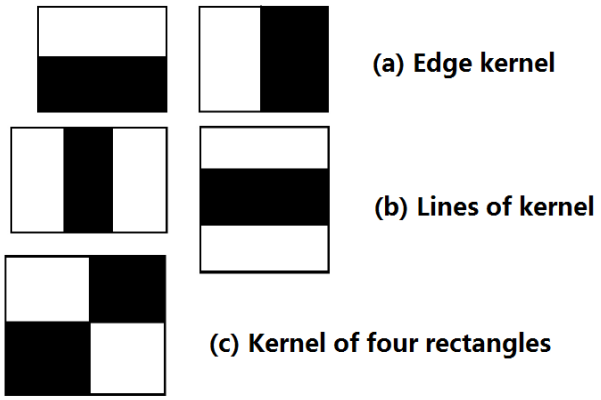

Figure 1. Core for Haar cascade.
Source: Adapted from mordvintsev (2013).

The core is a digital matrix representing the black-and-white part in **Figure 1**. Therefore, during scanning, each pixel in the image is multiplied by the corresponding number of cores. The difference between the white part pixels and the black part pixels produces characteristic values. At the end of projection, a set of features is obtained.

The second stage is based on AdaBoost, which aims to select a small part of visual features with higher intensity for a given image. These visual features are used for target detection and classification, which is based on pulse algorithm. Pulse is a machine learning technology that combines several weak classifiers to improve the overall accuracy[17].

The first selected feature is focused on the eye area, because the eye area is usually darker than the nose and cheek area. The second feature selected is based on the fact that the eyes are darker than the tip of the nose. For each visual feature, this method selects the resource with the lowest error rate to find parameters to classify future positive and negative faces.

The cascade process is carried out in the third stage, that is, the intermediate combination of degenerate tree classifiers. The strategy reduces the false positive rate by combining classifiers at each stage of the cascade. To illustrate this process, an image-based classifier is applied, which discards the images classified as negative, that is, the images without target features. The next step is to consider only the correctly classified images and apply another classifier by rejecting the negative images again. The cascade process continues with an ideal hit rate. **Figure 2**, adapted from Harmouch[18], illustrates the cascade classification process.

Similar to various Supervised learning methods, Haar also needs training stage. To do this, you must create an image library separated by positive (including the image of the object to be recognized) and negative (excluding them). At the end of the training process, the program stores the Haar characteristics of the object in an XML file, which will be used as the model of object detection.

In order to train the classifier, each image gets an equal weight at the beginning. After each classification, the weight of the incorrectly classified image will increase. Then, the same process is performed by calculating the error rate and the new

weight. Repeat this process until the required accuracy is reached or the required number of resources is found according to the error rate.

Haar Cascade is free and located in the OpenCV repository. It used to train classifiers with XML files. Therefore, its integration with the system is simple and flexible, which makes its implementation very popular in artificial vision projects for target detection.

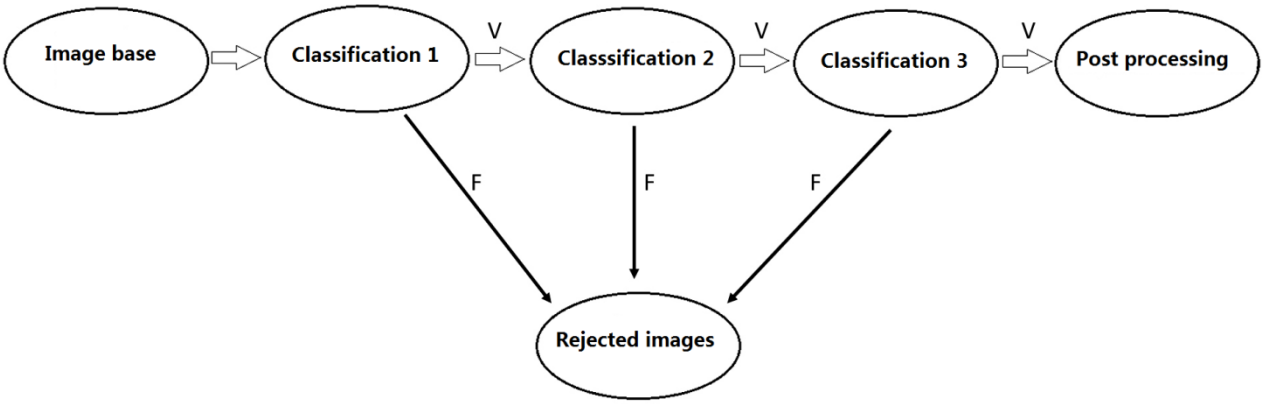

Figure 2. Cascade classification process In each step, a series of images (f) with wrong classification are rejected, and the image (V) with correct classification is then classified next time.
Source: Adapted from Harmouch[18].

3.2 Component diagram

Figure 3 shows the relationship between the application and the OpenCV framework responsible for image processing and coding[19]. Haar Cascade classifier is used to detect patterns in images. The benefits provided by Haar Cascade include ease of implementation and low consumption of computing resources to train and approve models.

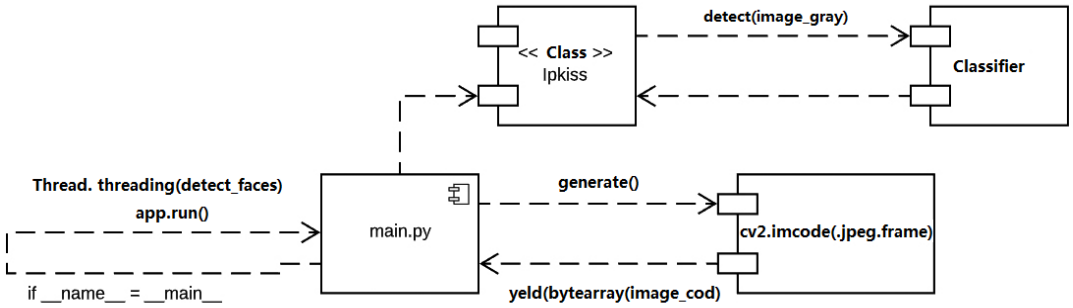

Figure 3. The component responsible for capturing and converting images.
Source: I designed it myself.

Figure 4 shows the connection between the application and MongoDB Atlas database[20]. This is a non-relational database, also known as NoSQL. This type of database improves the speed of transactions and does not give priority to the atomicity, consistency, isolation and persistence (ACID) related attributes in relational databases. MongoDB Atlas is a document oriented database that shares similar functions with other types of banks, such as securities, charts and columns. It also uses cloud computing resources. It is a service characterized by DaaS (database as a service), which allows any computer connected to the Internet to access its database management system and perform operations such as inserting, deleting, changing and querying records. The implementation model allows the flexibility of adjusting the computing resources used according to the needs of the solution, such as the transmission speed and the mode of storing data.

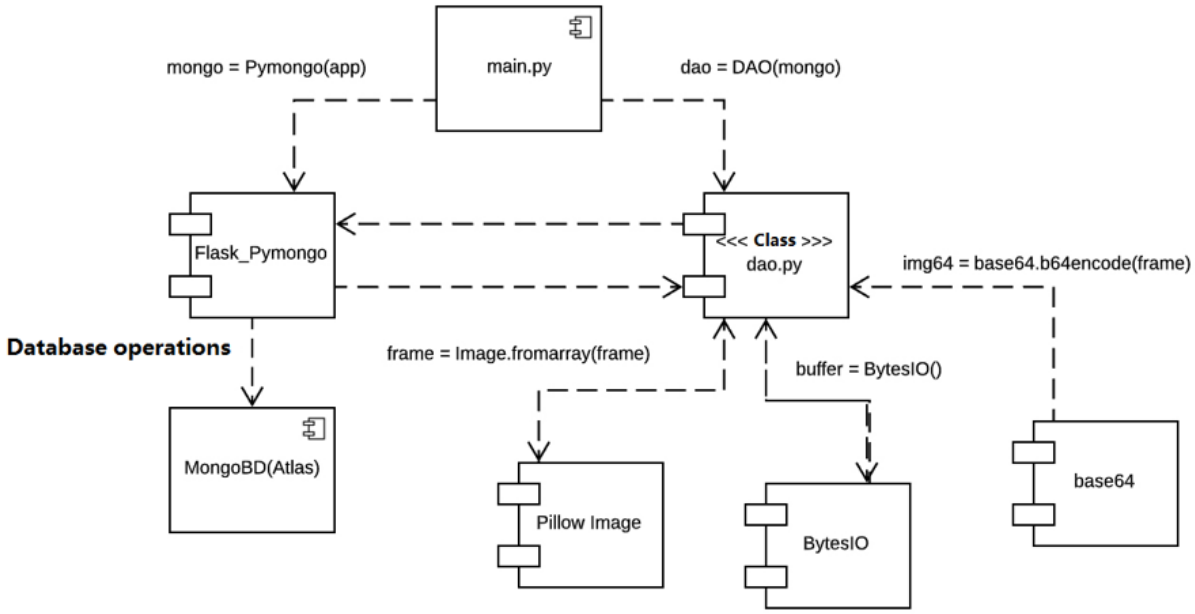

Figure 4. Transaction database component.
Source: I designed it myself.

PyMongo library[21]facilitates the interaction between python programming language and MongoDB database. It contains a set of tools to establish a connection between the display and data layers of a web application. The project uses PyMongo library to build the connection, query, change, insert and delete methods of records in the database.

The images taken by the camera are processed and then entered into the MongoDB Pillow database[22]. Then IO (Python Software Foundation, 2020b) and Base64 (Python Software Foundation, 2020a) were responsible for processing the task.

Figure 5 shows the relationship between the *front-end* application layer components developed in this project. Micro-*framework* Flask[23](Pallets,s.f.) is used to organize applications processed by applications. This *framework* was chosen because it has an optimized structure and sufficient resources to meet the needs of the project, and it adheres to the Python language. Another advantage is that it is modular and flexible, and can serve different types of web applications, from the simplest to the most complex.

Using Javascript libraries improves the usability of applications. Therefore, the jQuery Library[24] is used to create asynchronous licenses, so as to avoid loading pages multiple times during user interaction. Chartist.js library[25]. On the other hand, helps to build responsive and dynamic graphics, while the Materialize Library[26]was created by a group of students at Carnegie Mellon University according to Google's Material Design principles and is responsible for the style sheets of applications.

Algorithm 1 (**Table 1**) provides the functions of capturing, detecting and detecting human faces. The algorithm runs until the user exits the program (lines 2–14); take a camera image (line 3); resize to speed up the classification process (line 4); convert to grayscale (line 5) and create dark and clear areas to identify edge or texture changes; the date and time of the system in which the database record is stored (line 6); face detection using Haar features (line 7). If one or more faces are recognized (line 8), face cutting is performed in the original image (line 10) for each face detected in the image (lines 9 to 12), and the cut faces are stored in the database (line 11).

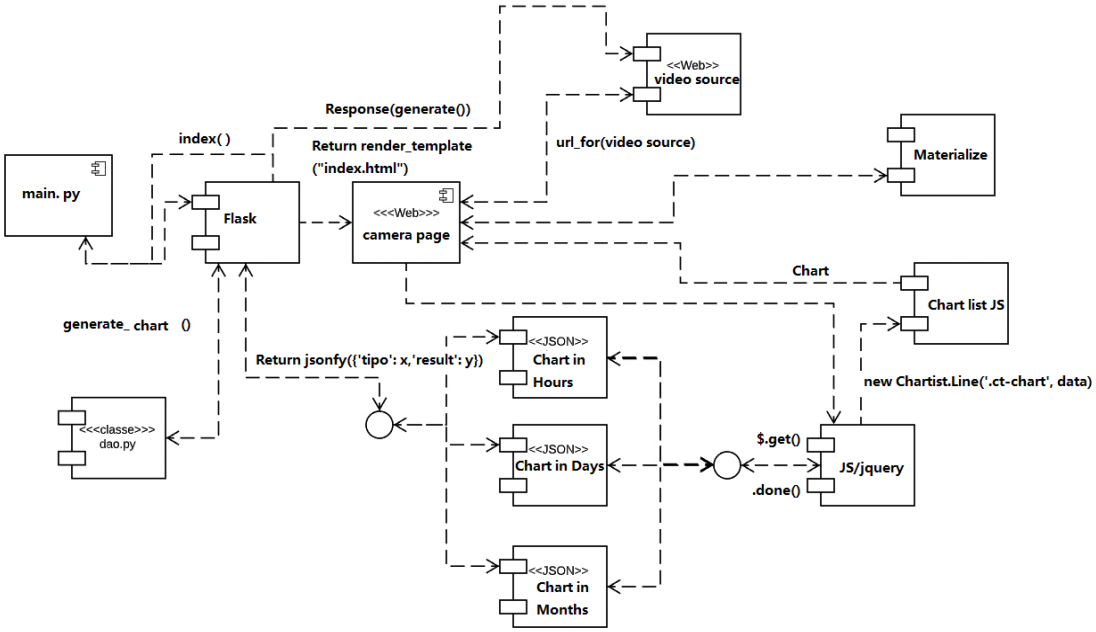

Figure 5. Components responsible for web visualization.
Source: I designed it myself.

Table 1. Algorithm 1 describes the process of detecting and storing faces without masks

Algorithm 1. Function detection and storage

1: **Function** detection
2: **As** long as the user does not complete **the program**
3: Capture new *img*
4: Resize *img*
5: *Convert* image to grayscale *imgg*
6: Store local *date and time*
7: Identify *faces* in *imgg* using Haar Cascade classifier
8: **If** the number of *faces* **>0, then**
9: **For** each *face* ∈ img **do**
10: Cut the face from the *img* and store it in *the face*
11: Stored *in* database
12: **End stop**
13: **End if**
14: **At the end**
15: **end**

Source: I designed it myself.

3.3 Technology used

Table 2 shows a summary of the technologies used in this project. The first column identifies the technology used, and the second column corresponds to the web application layer that requires the technology. When functions are performed on the user's machine, these layers are subdivided into clients, and when functions are performed on the server responsible for receiving requests, these layers are subdivided into servers. The last column provides the reasons for using each technology.

4. Results and discussion

A previously trained Haar Cascade algorithm model is used, which is not available in OpenCV library. The algorithm verification step implemented in the project uses a database containing 194 images (available from DataFlair, 2020). The images used are divided into two categories: one is the face image of people wearing personal protective masks, and the other is the face image of people without masks. Haar Cascade algorithm detects faces with or without protective masks through accuracy, revocation

and f_β-*score* measurement[27].

Table 2. Overview of technologies used

Technology	Layer	Reason
Git	The server	Version control code, save the implementation to. Warehouse. Is required to store items.
Python.	The server	It needs to be integrated with artificial vision and intelligent library.
OpenCV，	The server	Computer vision main library; In all the functions she performs, she is responsible for communicating with cla-actor.
Harr Cascade	The server	A classifier for detecting the presence of faces. A video frame.
Pillow	The server	Data needs to be written to temporary storage.
BytesIO	The server	Temporary storage needs to be created.
Base64	The server	In any server or web application, you need to convert a set of bytes into understandable text.
ImUtils	The server	It needs to capture the camera's IM, agen and cam, biar Tam, ano del m, arc.
Flask	The server	Micro framework for web development.
HTML	Customer	P> Atr6n web page.
Javasctipt	Customer	It needs to execute requests from web applications to servers and databases.
jQuery	Customer	It needs to simplify javascript functions.
Chartist JS	Customer	It needs to create simple, easy to accept graphics.
Materialize	Customer	Responsible for the layout generated by the style sheet.
PyMongo	The server	It is necessary for the application to interact with mongodb.
MongoDB Atlas	The server	Non relational and document oriented database management system.

Source: I designed it myself.

Table 3. Matrix confusion

		Truth value	
		Positive class	**Negative class**
Forecast	Positive class	90 （TP）	54 （FP）
	Negative class	7 (New)	43 (Tennessee)

Source: I designed it myself.

The precision, reversal and f_β-*score* were based on the values obtained by the classifier from the occurrences in measures of true positives (TP), false positives (FP), true negatives (TN) and false negatives (FN). The appearance belonging to TP category refers to the correct classification number of images with unmasked faces; FP value refers to the case of misclassification in the face image without mask; TN value refers to the incidence of correct classification of face and mask images, and the last FN value refers

to the incidence of incorrect classification of face and mask images. **Table 3** shows the confusion and distribution matrix of these values.

From these values, the accuracy is determined by the calculation of equation 1, where TP is divided by the sum of TP and FP:

$$P = \frac{TP}{TP + FP} \quad (1)$$

Conversely, revocation is defined by dividing the TP evaluation by the sum of TP and FN, as shown in equation 2:

$$R = \frac{TP}{TP + FN} \quad (2)$$

Conversely, the f_β*-score represented* in equation 3[28] is the harmonic average between precision and revocation, and its values represent parameters P and R, respectively. Parameter β can *be* used to assign different weights to the metrics used in the equation. In the performance evaluation used in this project, β value is equal to 1; therefore, accuracy and updating are equally important.

$$f_\beta \, \text{scor}\, e = \frac{\left(1+\beta^2\right)\cdot(P\cdot R)}{\left(\beta^2\cdot P + R\right)} \quad (3)$$

Therefore, through equations 1, 2 and 3, the precision, cancellation and f_1*-score* values shown in **Table 4** can be obtained.

Table 4. Results the accuracy, recall and f_1*score* were obtained

Exact	Revoke	f1-score
0.63	0.93	0.75

Source: I designed it myself.

In addition to evaluating the Haar Cascade algorithm, a minimum viable product (MVP) was constructed using the techniques shown in **Table 1**. The functional prototype of this web application demonstrates the performance of the technology and *hardware* evaluated in this project in a coupled and synchronous manner. **Figure 6** shows the initial screen for developing the prototype. The area marked with blue dotted line displays the video captured by the camera in real time and sends it to the computer vision algorithm. The second area is light green, showing a data carousel with the faces of people without protective masks extracted from the video. The third area, represented by orange dots, shows a graph of the number of individuals who did not wear masks during a specific period (monthly, weekly or daily).[1]

4.1. Discussion

This section discusses the test results, because if copied with other technologies, the results may be different, depending on whether these technologies are appropriate.

First of all, it must be remembered that hyperparameters are sensitive to changes because they control the classification process of images. Therefore, in order to optimize the classifier, you need to explore the parameter space by filtering the parameter space more accurately, which may take hours or even days to find the best super parameter option for your problem.

For example, the detectMultiScale function has two hyperparameters that directly influence the classification: ScaleFactor and MinNeighbours. The first is to determine the size of the image to identify more PE queue objects (in this case, for face detection); this parameter will affect the detection of people far away from the camera, but on the contrary, the false alarm rate and detection speed tend to increase. The second parameter is related to the kernel size (the matrix of adjacent pixels calculated and analyzed), which makes the image clearer and the details more related to the classifier; However, this will require greater processing power.

With regard to image processing, it should be noted that the performance of the classifier may be different by applying other technologies and filters

[1] MVP demo video can be obtained from the following link: https://youtu.be/QmCs_piHZkw. Accessing the solution repository https://github.com/Eskandar1/Ipkiss

provided by OpenCV. For example, it depends on whether the image is balanced or binarized by Gaussian smoothing. These technologies and filters have their own super parameters for optimization.

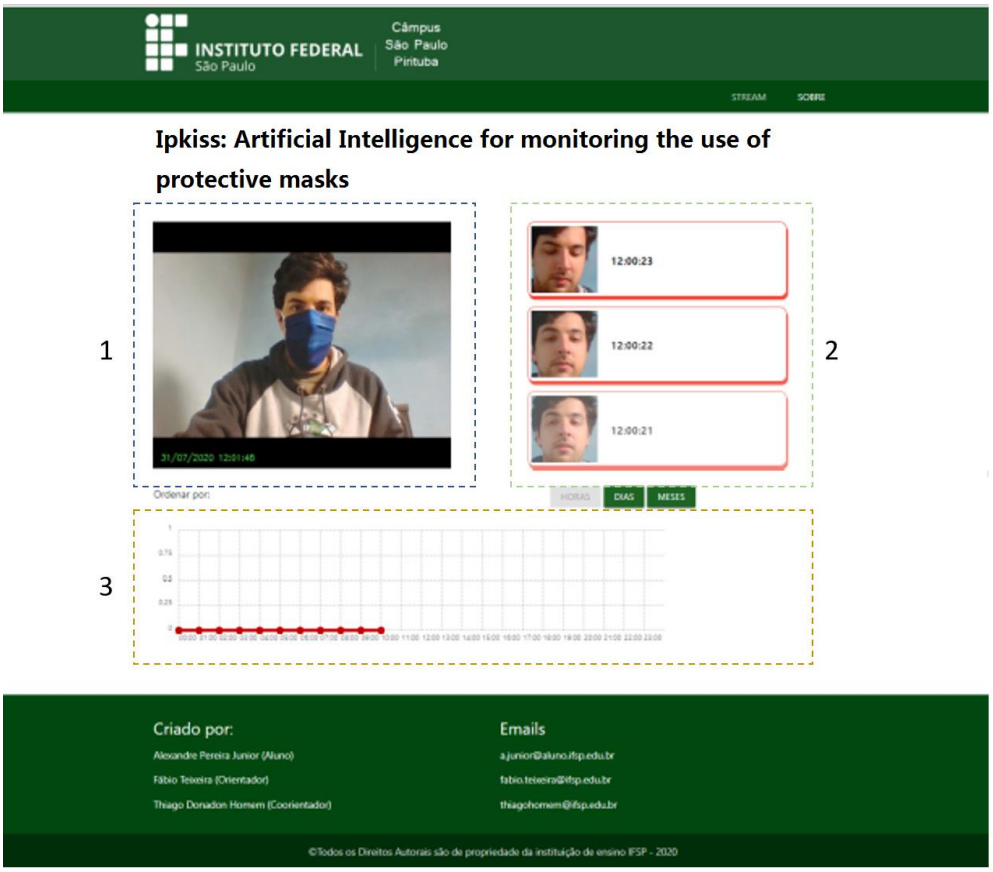

Figure 6. A web prototype system is developed.
Source: I designed it myself.

Another thing worth emphasizing is that the accuracy of f_1-*score* formula is as important as revocation. In addition to accuracy, the algorithm must also detect the maximum number of faces without protective masks in a given scene, which is represented by the f_1-*score* performance measure achieved by the project through the balance between accuracy and revocation. This allows good results to be obtained by MVP analysis of a large number of personnel movements.

One limitation of Haar Cascade classifier is that it can only recognize positive faces. In an uncontrolled environment, people can walk in different directions, so they won't detect faces that don't look at the camera. One way to solve this limitation of the classifier is to use a model with deep learning technology, which contains hierarchical separated neurons, which are specially used for complex pattern recognition in the training process, such as recognizing protective masks, regardless of facial angle.

The data processed during prototype execution is stored in the database. This storage allows them to cross with other sources; for example, public data provided by Datasus, the Information Technology Department of Brazil's single health system (SUS). Such data cross cutting can enhance the analysis of the epidemic, including virus transmission and the use of protective masks.

5. Experiment

Considering the prevalence of COVID-19 and the nursing needs for virus spread, this paper introduces the application of computer vision technology to monitor the use of mask. A web application is developed, which can detect and record the facial images of people without additional wrist guards. This is an application that can be easily implemented on low-cost devices.

The system can detect the face without protective mask with 63% accuracy and record it in the database. This enables those responsible for monitoring the specific environment of implementation to meet the need for effective measures to combat and prevent infectious diseases.

As a suggestion for future work, we intend to use *deep learning* technology to improve the efficiency of the solution. This recommendation must weigh the relationship between facial accuracy and performance so that the application can be implemented on low-cost devices.

Conflict of interest

The authors claim that there is no potential con-flict of interest related to this article.

Acknowledgement

The authors claim that, in order to implement this article, they received financial support from the Institutional Program of Scholarships for Scientific and Technological Initiation of the Federal Institute of Education, Science and Technology of São Paulo.

The authors wish to thank the Federal Institute of Education, Science and Technology of São Paulo (IFSP) for their support in the development of this work. Alexandre Pereira Junior thanks the Institutional Program of Scholarships for Scientific and Technological Initiation of the IFSP for its support in the realization of this article.

References

1. Vieira J, Ricardo O, Hannah C, et al. What do we know about COVID-19? Revista Da Associação Médica Brasileira 2020; 66 (4): 534–540.
2. Reis F. Esteudo avalia efeka Das mascados de prot é no coronaverus. Pfarma; 2020.
3. Hewings-Martin Y. How do SARS and MERS compare with COVID-19? Medical News Today; 2020.
4. Diário Oficial de São Paulo. Decreto 59396 2020 de São Paulo SP; 2020. Available from: https://bit.ly/3hb-hJGbb.
5. Sistema de Monitoramento Inteligente do Governo de São Paulo (SIMI-SP). Isolamento 2020. Governo do Estado de São Paulo. Available from: https://www.saopaulo.sp.gov.br/coronavirus/isolamento/.
6. Pontes R. Inteligência artificial nos investimentos. Clube de Autores (managed); 2011.
7. Backes AR, Junior JJ. Introdução à visão computacional usando Matlab. Alta Books Editora; 2019.
8. Almanza JC, Visão computacional e aprendizagem automática para aplicações em agropecuária e ciências forenses; 2018. Available from: http://www.gpec.ucdb.br/pistori/orientacoes/planos/carlos2018.pdf.
9. Kissler SM, Tedijanto C, Goldstein E, et al. Projecting the transmission dynamics of SARS-CoV-2 through the postpandemic period. Science 2020; 368(6493): 860–868.
10. Portal do Governo. Governo de SP apresenta Sistema de Monitoramento Inteligente contra coronavírus. Governo do Estado de São Paulo; 2020. Available from: https://bit.ly/2X3ReFn
11. Araujo ZL,Rrosito J C. The deep neural network is used to detect the gender in wild animals in real time. In 2018, the 31st sibrapi graphics, patterns and images Conference (sibrapi). 2018. p. 118-125. https://doi.org/10.1109/sibgrapi.2018.00022.
12. Botelho G, Papa J, Marana A. On learning the deep local features of robust face deception detection. In 2018, the 31st sibrapi graphics, patterns and images Conference (sibrapi). 2018. p. 258-265. Https://doi.org/10.1109/sibgrapi.2018.00040.
13. Finizola JS, Tagino JM, Teodoro FG, Lima CA. (2019). A comparative study of deep face, autoencoder and traditional machine learning technology for biometric face recognition. 2019 International Joint Conference on neural networks (IJCNN). 2019. p. 1–8.
14. Mordvintsev A. Face detection using Haar cascade. OpenCV-Python Tutorials 2013.

15. Viol P, Jones M. Enhanced cascading fast object detection using simple features. Proceedings of the 2001 IEEE Computer Society Conference on computer vision and pattern recognition. CVPR 2001. Https://doi.org/10.1109/CVPR.2001.990517.
16. Lienhart R, Maydt J. An extended set of Haar-like features for rapid object detection. En Proceedings. International Conference on Image Processing 2002. Available from: Https://doi.org/10.1109/ICIP.2002.1038171
17. Duarte JC. O algorithm enhancement at startup and its application. Coleção Digital 2009.
18. Harmouch M. The Haar cascade classifier in opencv intuitively explains this. Medium 2020.
19. OpenCV Team. Opencv 2020. Available from: Https://opencv.org/.
20. MongoDB Inc..Montgobu atlas 2020. Available from: Https://www.mongodb.com/cloud/atlas
21. Dirolf, M. Pimongo: Python driver for mongodb (3.11.0) [software]. Python 2020. Available from: https://pypi.org/Project/pimongo/.
22. Clark A, Lundh F, GitHub contributors. Pillow: Python Imaging Library (7.2.0) [software]. Python 2020. Available from: https://python-pillow.org.
23. Pallets (s. f.). Welcome to Flask's documentation [página web]. Consultado el 14 de agosto de 2020. Available from: https://flask.palletsprojects.com/en/1.1.x/.
24. OpenJS Foundation. Jquery 2020. Available from: Https://jquery.com/.
25. Kunz G. Chart. Chartist.js Simple responsive charts 2019. Available from: Https://gionkunz.github.io/chartist-js/.
26. Chang A, Wang A, Mark A, et al. Documentation [software]. Google 2020. Available from: Https://materializecss.com/.
27. Gehanno JF, Rollin L, Le Jean T, et al. Determine the accuracy and recall of the search strategy of MEDLINE rework study. Journal of Occupational Rehabilitation 2009. 19 (3):223–230.
28. Magdy W, Jones GJ. Chairman: Evaluate the scoring criteria for recall oriented information retrieval applications. Proceedings of the 33rd ACM SIGIR International Conference on research and development of information retrieval. 2010. p. 611–618.