v. 3, n. 27, 2023

Journal of Engineering Research

COMPARISON OF TOOLS TO LOCATE OCCLUDED FACES

Jose Omar de Jesus Trujillo Quintero
Tecnológico Nacional de México/CENIDET, Cuernavaca, Morelos, Mexico

Andrea Magadán Salazar
Tecnológico Nacional de México/CENIDET, Cuernavaca, Morelos, Mexico

Jonathan Villanueva Tavira
Universidad Tecnológica Emiliano Zapata del Estado de Morelos, Mexico

Raúl Pinto Elías
Tecnológico Nacional de México/CENIDET, Cuernavaca, Morelos, Mexico

Abstract: Derived from the pandemic, the World Health Organization (WHO) recommends using face masks as a way to contribute to reducing the spread of the virus that causes COVID-19; however, the use of the face mask causes many current facial biometric systems to be inefficient regarding the location and recognition of people when their faces are occluded with the face mask. This article presents a comparison between four of the main facial locators most used in people identification systems with pre-trained models, but applied to the detection of faces with occlusion caused by the use of a mask in uncontrolled environments. The tools that were evaluated are Dlib, MTCNN, DNN Facial Detector and MediaPipe. For evaluation, a sample of images was drawn from the public sets MaskedFace-Net, MFDD, RMFRD, SMFRD, and a small proprietary set with lighting changes, rotation, and various types of occlusion. After experimentation it was found that MediaPipe was better obtaining a precision of 97.5%, an accuracy of 86.7% and a 92.9% f1 score in this work.
Keywords: Location, Occluded Faces, Evaluation, Face Mask Detection, Artificial Vision.

INTRODUCTION

Derived from the pandemic generated by the virus that causes COVID-19, the World Health Organization (WHO) recommends that all people constantly wear masks in public, since it has been shown that the use of a face mask plays an important role in preventing the spread of the coronavirus (Organization & others, 2020). For this reason, work continues on the development of people recognition systems considering face occlusion to update biometric access systems and video surveillance systems for the search and identification of criminals.

The development of systems that consider the occlusion of the face has been worked on for a long time as a complement to facial recognition, since criminals take advantage of this weakness of facial recognition systems to commit robberies and other actions against the law. However, despite the advances, one of the main challenges continues to be locating the face in uncontrolled environments with changes in pose, scale, rotation, and light intensity, considering that approximately half of the face is occluded by the mask.

In order to determine which localization tool of those existing in the literature has a better performance against the conditions listed above, a review was first carried out of the state of the art finding that (Wang et al., 2020) uses the Dlib library (Dlib, 2022) to develop a system with which, based on images of faces without occlusions, it locates the face and adds the mask automatically. They called this dataset SMFRD and it can also be used to detect the mask (Wang et al., 2020). In the same article, they develop a system that reports 95% accuracy with a multigranularity masked facial recognition model.

Due to the good performance of deep learning in various Computer Vision applications, various models for face detection have been proposed. In the state of the art there are several proposals that report high yields; for example, in (Nagrath et al., 2021) they make use of the OpenCV DNN module containing the Single Shot Multibox Detector (SSD) model (Liu et al., 2016) and ResNet-10 (Anisimov & Khanova, 2017) as the main architecture to work in a real-time approach to detection.

Another proposal to locate occluded faces is the implementation of CNN architectures, for example, in (Wan & Chen, 2018) they propose a deep learning model, which they called MaskNet. They used the CASIA-Webface dataset (Yi et al., 2014) for training and the AR dataset (Martinez & Benavente,

1998) for evaluation. As processing, they perform the alignment and resizing of the dataset images with MTCNN (Zhang et al., 2016), it is mentioned that taking into account only faces with sunglasses, they obtained an accuracy of 90.9%.

In another application using CNNs (J. Barrios, 2020) they developed an algorithm for the detection of mask use in real time, which can be implemented using a PC and a conventional webcam, thanks to the use of tools such as OpenCV, Keras and Python's TensorFlow. It is also mentioned that 3835 images with a mask and without a mask were used for training. In a similar way (Cayetano, 2021) mentions that they developed a system based on Deep Learning to detect the use of face masks automatically, they also created their own dataset consisting of 700 images with and without face masks that were taken in the surroundings of where the author lives, taking into account age diversity, skin tones, and lighting changes according to the time the images were captured.

The innovation of facial recognition and identification systems are getting better and better thanks to the investments and development generated by large companies such as Google who implemented a development framework called MediaPipe. This tool not only focuses on the location of the face, but also has different types of solutions for the location of the human body, hair, etc., as explained in (Kukil, 2022).

The objective of this article is to present a comparison of the four main tools for face detection in an image or video in real (uncontrolled) environments, especially considering their behavior by not detecting important reference points such as the nose, mouth and chin. It is important to mention that the selected tools were implemented using their pre-trained models for face detection.

The rest of the work is organized as follows: section 2 describes the tools implemented to localize the face. Section 3 presents the experimentation carried out, this section also shows the information on the sets of images considered, the metrics used to evaluate the performance of each tool, and the results obtained are discussed. Finally, section 4 contains the conclusions of the article and the future work to be carried out.

DETECTION TOOLS

There are several proposals for the detection of the face in an image, capable of locating all the faces present simultaneously; however, Dlib, MTCNN, DNN Facial Detector in OpenCV were chosen as they are the most used in the literature and with respect to MediaPipe it is a relatively new tool that has achieved great importance thanks to its multiple applications on various platforms, as well as being free and easily accessible. The four tools present pre-trained models to locate the face, which does not require additional training.

DLIB

Dlib (Dlib, 2022) is a modern toolset containing machine learning algorithms to create complex software in the C++ language to solve real-world problems. It is used in both industry and academia in a wide range of domains and can be used on a variety of platforms, including Python.

Dlib integrates with HOG + Linear SVM face detector which is fast and efficient. But, due to the nature of how the Histogram of Oriented Gradients (HOG) descriptor works, it is not invariant to changes in rotation and viewing angle (Rosebrock, 2021).

Additionally, the Dlib library has a pre-trained facial reference detector, which is used to estimate the location of 68 coordinates (x, y) that are assigned to facial structures or elements, as can be seen in figure 1 (Rosebrock,

2017).

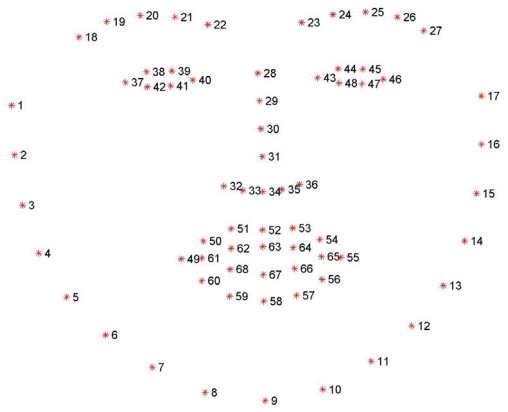

Fig 1 Dlib Face Dots Template (Rosebrock, 2017).

MTCNN

They combine cascaded CNNs by multi-task learning (Zhang et al., 2016), that is, they implement a deep cascaded multitasking structure making use of the inherent correlation between them to increase performance. Each of the cascading layers consists of three stages of carefully designed deep convolutional networks, predicting facial landmarks from a coarse to a fine shape. In addition to detecting the location of the face, it aligns the important parts of the face, a necessary step especially in real environments.

MEDIAPIPE

MediaPipe Face Detection (MediaPipe, 2020) is an ultra-fast face detection solution that comes with 6 waypoints and multiple face support. It is based on BlazeFace (Bazarevsky et al., 2019), a lightweight and well-performing face detector designed for mobile GPU inference. The detector's super real-time performance enables 3D facial key point estimation (for example, MediaPipe Face Mesh that has 468 facial points, as can be seen in figure 2).

Also, because it is a cross-platform tool, the developer can configure the application built with MediaPipe to manage resources efficiently (for both CPU and GPU) to achieve low latency performance, to handle synchronization of time series data as well as audio and video frames, and to measure performance and resource consumption (Lugaresi et al., 2019

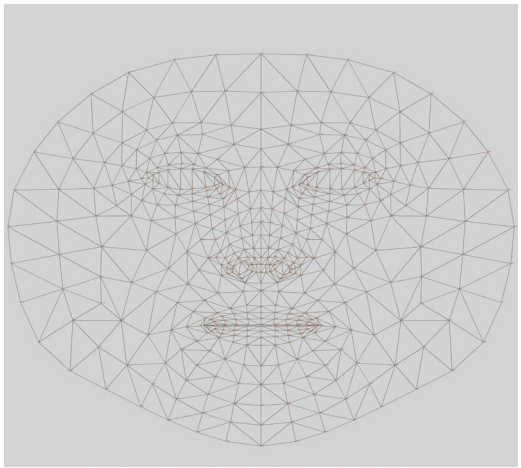

Fig 2 MediaPipe Facial Dots Template.

DNN FACE DETECTOR IN OPENCV

OpenCV's DNN contains the Single Shot Multibox Detector (SSD) model (Liu et al., 2016) and ResNet-10 (Anisimov & Khanova, 2017) as the main architecture to work in a real-time approach to face detection (Nagrath et al., 2021).

DNN requires two Caffe files, which are deploy.prototxt which defines the network architecture and res10_300x300_ssd_iter_140000.caffemodel contains the layer weights (Kavitha et al., 2021), these files can be downloaded from the Github repository

EXPERIMENTATION

To carry out the evaluation of the tools considering real variations of the environment, a sample of four sets of public images was taken. These images feature faces covered with masks, complex backgrounds, different skin tones, scale, perspective, age, ethnicity, and the presence of one or more faces in the image. The faces can also contain accessories such as

glasses, hair, caps, scarves and obviously face masks of different colors. In addition, an own set was created with variations of the type of face mask and presence of occlusion on the face. The datasets are described below:

DATASETS

MaskedFace-Net (Cabani et al., 2021): The dataset has more than 137,016 images of masked faces organized into two subsets:

- The set of correctly masked faces (covering the nose and mouth) (called CMFD), which integrates 49% of the images.
- The set of images of incorrectly masked faces (called IMFD) corresponds to 51% of the images.

This dataset was built by adding a simulated mask using the FFHQ dataset due to its wide variety in terms of age, ethnicity, point of view, lighting, and image background.

Examples of the images can be seen in figure 3:

Fig 3 Image sample of the two subsets of MaskedFace-Net (Cabani et al., 2021).

MFDD (Wang et al., 2020): Set called "Masked Face Detection Dataset" (MFDD). It contains 24,771 images of masked faces which are from the Internet and other related works, therefore the dataset contains a high variability.

RMFRD (Wang et al., 2020): Real World Masked Facial Recognition Dataset (RMFRD). It includes 5,000 photographs of 525 people wearing masks and 90,000 images of the same 525 subjects without masks. It is mentioned that the images contain the front view faces of public figures, downloaded from massive Internet resources, therefore, the images present high variability.

SMFRD (Wang et al., 2020): Simulated Masked Face Recognition Dataset (SMFRD) built from the LFW (Huang et al., 2008) and Webface (Yi et al., 2014) datasets, building a set of simulated masked faces of 500,000 images from 10,000 subjects that can be used with their original counterpart. no mask.

Figure 4 shows an example of the images of the MFDD, RMFRD, and SMFRD datasets.

Fig 4 Example of image datasets: MFDD in the first row, RMFRD in the second row, and SMFRD in the third row (Wang et al., 2020).

Own set: It consists of 60 images acquired on different days, which causes lighting changes, face rotation and different types of occlusion, in addition to having greater variability in the type of mask in real conditions, as can be seen in figure 5.

Fig 5 Sample of own dataset images

From the aforementioned datasets, a sample was taken from each one with a total of 130 images. The main aspect to evaluate is the robustness of the tools to locate a face in the image when, in addition to the aforementioned conditions, the face is occluded by hair, glasses, a cap, and approximately half of it is occluded by the use of a face mask.

EVALUATION METRICS

To evaluate the performance of the tools considered in face localization, the classic metrics in the area of machine learning are used, in this case they are: accuracy, precision, sensitivity and F1-score. These metrics use the confusion matrix as a basis. Each column of the matrix represents the number of predictions of each class, while each row represents the instances in the actual class as shown in Figure 6. In practical terms, the matrix allows you to see the hits and misses of the model. (J. I. Barrios, 2019).

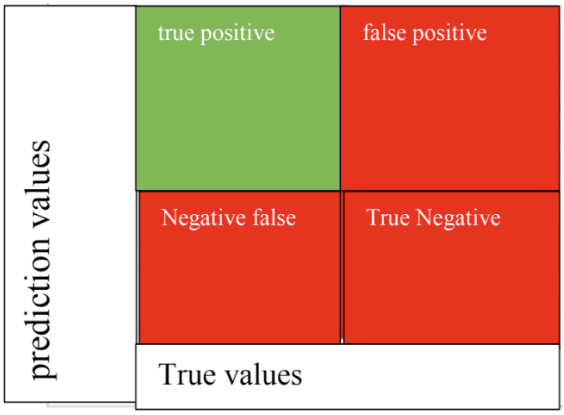

Fig 6 Confusion matrix (J. I. Barrios, 2019).

$$\text{Accuracy} = \frac{Vp + Vn}{(Vp + Fp + Fn + Vn)} \quad (1)$$

$$Precision = \frac{Vp}{(Vp + Fp)} \quad (2)$$

$$Sensibility = \frac{Vp}{(Vp + Fn)} \quad (3)$$

$$f1\ score = 2 * \frac{Sensibility * Precision}{(Sensibility + Precision)} \quad (4)$$

IMPLEMENTATION OF THE TOOLS

The implementations were made in Python 3.9, using the Dlib library, the MTCNN library, the imutils library together with Mediapipe and finally for DNN two additional files are required, deploy. prototxt.txt that defines the architecture and res10_300x300_ssd_iter_140000.caffemodel that defines the weights of the pre-trained network corresponding to the Caffe model. The OpenCV library is also used to read and analyze the images.

Each input image is processed with the four tools. Each one of them locates the potential area where the face is located and searches for the referential points. When the algorithm reaches a certain certainty value, it draws a rectangle and provides the face information (positions) in detail.

Since the images considered from the different datasets had a different dimensionality from each other, it was necessary to resize (normalize) them to 300x300 pixels, this helps their visualization.

The images were also renamed to have uniformity in the name and that the system can read and load them automatically. No other preprocessing was carried out on them, since the objective of the work is to check the robustness of each tool with the original parameters of their installation.

TESTS

As it was already mentioned, the objective of the experimentation is to evaluate which of the main and most used tools available in the literature for face detection and localization is more robust in real applications whose images present faces in uncontrolled environments; that is, with different scale, rotation, light intensity and, above all, evaluate its ability to treat occlusion caused by garments or clothing accessories such as hats, caps, helmets, prescription glasses, sunglasses, hair, even

makeup, without forgetting the occlusion of the face caused by the use of face masks.

ANALYSIS OF RESULTS

Table 1 shows the results obtained by each tool, in the metrics considered.

Metrics	Tools			
	Dlib	DNN	MediaPipe	MTCNN
Accuracy	0.530	0.559	**0.867**	0.373
Precision	**1.00**	0.564	0.975	**1.00**
Sensibility	0.530	**0.984**	**0.886**	0.373
F1 Score	0.693	0.718	**0.929**	0.543

Table 1 Results obtained.

As it can be seen in table 1, Dlib and MTCNN have good precision thanks to the fact that they perfectly locate all the frontal faces present in the image. The correct detection is carried out even in the presence of the use of face masks, the use of a helmet, the presence of hair on the forehead, the use of prescription or sun glasses and the use of hats. It does not generate false positives, even in images without the presence of faces or in faces that cannot be detected due to conditions due to changes in lighting, scale, and average rotation. Actually, the main factor limiting their performance is that both tools need to be able to locate both eyes completely, therefore they score poorly in the accuracy, sensitivity and F1 Score metrics.

For their part, DNN and MediaPipe are the most efficient tools that generate a good response to the aforementioned environmental conditions, for face detection with occlusion they obtain 0.984 and 0.886 sensitivity respectively. The MediaPipe tool has the best result of 0.867 accuracy over the other three tools when detecting the face, even when there are rotations that limit the complete presence of the parts of the face.

Figure 7 shows examples of the detection carried out with the 4 analyzed tools, the image on the upper left was obtained using the Dlib face detector, the upper right image was obtained with the DNN face detector, the lower left image was obtained with the MediaPipe detector and finally the lower right image was obtained with the MTCNN detector.

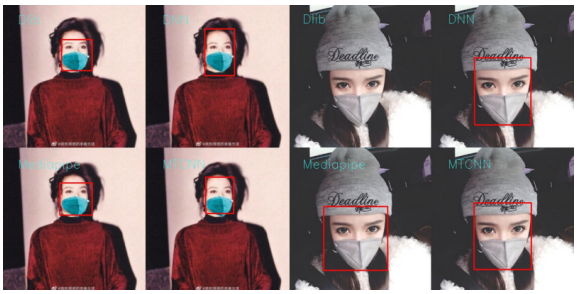

Fig 7 Sample of the results obtained with each facial detector.

The four tools work well with changes in the scale, locating faces present in small regions and in medium-sized regions (considering the dimension of the entire image). However, MediaPipe tends to have problems with small faces (further away in the image) but better detects faces with a higher appreciation of rotation. On the other hand, the DNN tool generates false positives if the face almost completely covers the entire corresponding region of the image to be analyzed; that is, it perfectly locates the face in the image, but it also generates a rectangle in the lower right part of the image, that is, it presents false positives, as shown in figure 8.

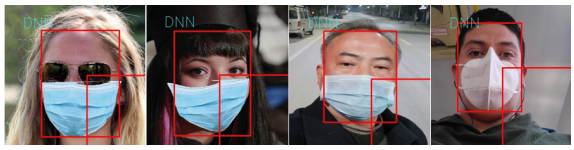

Fig 8 Visualization of False Positives with DNN.

Finally, it was observed that MediaPipe is more robust in detecting faces with occlusions, with the mask being the largest occlusion present in the image, it is also capable of

detecting the face in extreme conditions of low lighting and greater rotation. In addition, its 468 facial point mesh helps to locate the face in a general way, as well as to segment the different elements in detail without problem, which means that the description of the face can be used in various applications in recognition systems.

CONCLUSION

Derived from the global pandemic by COVID-19, facial recognition systems have to be updated, the first stage being the location of occluded faces in real environments. In this article, four free and public tools were reviewed, which have a pre-trained model to detect and locate the face in real environments. In addition, to generate a new dataset with faces covered with face masks, which facilitates the development of new systems.

It can be concluded that Dlib is one of the most widely used tools in the literature, having good results in detecting faces without the presence of occlusion, but it has problems with low light images and when it cannot locate both eyes. In the case of DNN, it generates false positives when the face almost completely covers the image; MTCNN does not feature false positive detection; however, it is the least accurate in detecting the face when there is some type of occlusion such as the use of a face mask. For its part, MediaPipe is a relatively new tool with sufficient robustness to perform the task of detecting and locating the occluded face and in uncontrolled environments, obtaining a precision of 97.5%, an accuracy of 86.7% and a 92.9% f1 score in this work. As future work, it is intended to compare the tools analyzed in this article and other tools that require training prior to their implementation.

THANKS

To the National Council of Science and Technology (CONACYT) for the scholarship granted to carry out my master's studies at TecNM/CENIDET.

REFERENCES

Anisimov, D., & Khanova, T. (2017). Towards lightweight convolutional neural networks for object detection. *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 1–8.

Barrios, J. (2020, June 7). *Algoritmo para la detección de mascarilla*. https://www.juanbarrios.com/algoritmo-para-la-deteccion-del-uso-de-la-mascarilla/

Barrios, J. I. (2019, June 26). *La matriz de confusión y sus métricas – Inteligencia Artificial –*. https://www.juanbarrios.com/la-matriz-de-confusion-y-sus-metricas/

Bazarevsky, V., Kartynnik, Y., Vakunov, A., Raveendran, K., & Grundmann, M. (2019). *BlazeFace: Sub-millisecond Neural Face Detection on Mobile GPUs*. 3–6. http://arxiv.org/abs/1907.05047

Cabani, A., Hammoudi, K., Benhabiles, H., & Melkemi, M. (2021). MaskedFace-Net – A dataset of correctly/incorrectly masked face images in the context of COVID-19. *Smart Health*, *19*, 100144. https://doi.org/10.1016/j.smhl.2020.100144

Cayetano, I. R. P. (2021). Detección automática de rostros con cubreboca o sin cubreboca para restringir el acceso a institución educativa. *Revista Aristas*, *8*(16), 154–160.

Dlib. (2022, January 24). *dlib C++ Library*. http://dlib.net/

Huang, G. B., Mattar, M., Berg, T., & Learned-Miller, E. (2008). Labeled faces in the wild: A database forstudying face recognition in unconstrained environments. *Workshop on Faces in'Real-Life'Images: Detection, Alignment, and Recognition*.

Kavitha, K. R., Vijayalakshmi, S., Annakkili, A., Aravindhan, T., & Jayasurya, K. (2021). Face Mask Detector Using Convolutional Neural Network. *Annals of the Romanian Society for Cell Biology*, 1979–1985.

Kukil. (2022, March 1). *Introducción a MediaPipe | AprenderOpenCV*. https://learnopencv.com/introduction-to-mediapipe/

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. C. (2016). Ssd: Single shot multibox detector. *European Conference on Computer Vision*, 21–37.

Lugaresi, C., Tang, J., Nash, H., Mcclanahan, C., Uboweja, E., Hays, M., Zhang, F., Chang, C., Yong, M. G., Lee, J., Chang, W., Hua, W., Georg, M., & Grundmann, M. (2019). *MediaPipe : A Framework for Perceiving and Augmenting Reality*.

Martinez, A., & Benavente, R. (1998). The AR face database, CVC. *Copyright of Informatica (03505596)*.

MediaPipe. (2020). *Face Detection - mediapipe*. https://google.github.io/mediapipe/solutions/face_detection

Nagrath, P., Jain, R., Madan, A., Arora, R., Kataria, P., & Hemanth, J. (2021). SSDMNV2: A real time DNN-based face mask detection system using single shot multibox detector and MobileNetV2. *Sustainable Cities and Society*, *66*, 102692.

Organization, W. H., & others. (2020). *Advice on the use of masks in the context of COVID-19: interim guidance, 5 June 2020*.

Rosebrock, A. (2017, April 3). *Facial landmarks with dlib, OpenCV, and Python - PyImageSearch*. https://pyimagesearch.com/2017/04/03/facial-landmarks-dlib-opencv-python/

Rosebrock, A. (2021, April 19). *Detección de rostros con dlib (HOG y CNN) - PyImageSearch*. https://pyimagesearch.com/2021/04/19/face-detection-with-dlib-hog-and-cnn/

Wan, W., & Chen, J. (2018). Occlusion robust face recognition based on mask learning. *Proceedings - International Conference on Image Processing, ICIP*, *2017-Septe*, 3795–3799. https://doi.org/10.1109/ICIP.2017.8296992

Wang, Z., Wang, G., Huang, B., Xiong, Z., Hong, Q., Wu, H., Yi, P., Jiang, K., Wang, N., Pei, Y., Chen, H., Miao, Y., Huang, Z., & Liang, J. (2020). Masked face recognition dataset and application. *ArXiv*, 1–3.

Yi, D., Lei, Z., Liao, S., & Li, S. Z. (2014). Learning face representation from scratch. *ArXiv Preprint ArXiv:1411.7923*.

Zhang, K., Zhang, Z., Li, Z., & Qiao, Y. (2016). Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, *23*(10), 1499–1503.