

DetECCIÓN DE MASCARILLAS UTILIZANDO RECONOCIMIENTO FACIAL

Facemask detection using face recognition

Xoán Teira¹, Nikolai Andreas Guerra², Jahir Gamar Castillo-Santamaria³, Lilia Muñoz⁴, Vladimir Villarreal⁵

Teira, X; Guerra, N.A; Castillo-Santamaria, J.G; Muñoz, L; Villarreal, V. Detección de mascarillas utilizando reconocimiento facial. *Tecnología en Marcha*. Vol. 36, número especial. Octubre, 2023. V Congreso Internacional en Inteligencia Ambiental, Ingeniería de Software, Salud Electrónica y Móvil. Pág. 57-65.

 <https://doi.org/10.18845/tm.v36i8.6928>

1 Universidad Tecnológica de Panamá, Panamá.

Correo electrónico: xoan.teira@utp.ac.pa

 <https://orcid.org/0000-0002-4216-0075>

2 Universidad Tecnológica de Panamá, Panamá.

Correo electrónico: nikolai.guerra@utp.ac.pa

 <https://orcid.org/0000-0002-0530-253X>

3 Universidad Tecnológica de Panamá, Panamá.

Correo electrónico: jahir5656@gmail.com

 <https://orcid.org/0000-0002-4693-8673>

4 Universidad Tecnológica de Panamá, Panamá.

Correo electrónico: lilia.munoz@utp.ac.pa

 <https://orcid.org/0000-0002-4011-2715>

5 Universidad Tecnológica de Panamá, Panamá.

Correo electrónico: vladimir.villarreal@utp.ac.pa

 <https://orcid.org/0000-0003-4678-5977>



Palabras clave

Reconocimiento facial; mascarillas; Python; OpenCV; Arduino; MediaPipe; Inteligencia Artificial.

Resumen

Este proyecto se realizó como respuesta a algunas exigencias actuales las cuales dictan que el uso de mascarillas es obligatorio dentro de espacios cerrados como Colegio y Universidades. El objetivo que se planteó fue realizar un prototipo que mediante el uso de una cámara y aplicando la inteligencia artificial se pudiera detectar si una persona está o no utilizando una mascarilla. Para el mismo se utilizaron algunas librerías de Python, como lo es media pipe también openCV, Arduino. Esta inteligencia artificial para su correcto funcionamiento fue entrenada con diversos videos de personas sin y con mascarillas y mediante el reconocimiento de patrones se llegó al objetivo, al detectar una persona sin mascarilla se prendera un led y se emitirá un sonido de alerta, por otro lado, en caso de que reconozca que la persona tiene la mascarilla solo se encenderá un led verde.

Keywords

Facial detection; face mask; Python; OpenCV; Arduino; MediaPipe; Artificial Intelligence.

Abstract

This project was carried out in response to some current demands which dictate that the use of masks is mandatory within closed spaces such as Schools and Universities. The objective that was raised was to make a prototype that, using a camera and applying artificial intelligence, could detect whether or not a person is wearing a mask. For it, some Python libraries were used, such as media pipe, also openCV, Arduino. This artificial intelligence for its correct operation was trained with various videos of people without and with masks and through pattern recognition the objective was reached, when detecting a person without a mask an LED will turn on and an alert sound will be emitted, on the other hand, in case it recognizes that the person has the mask, only a green LED will light up.

Introducción

El uso de mascarillas se ha vuelto cotidiano con la llegada de la pandemia del Covid-19 y el uso correcto es una de las principales medidas de seguridad.

Aun sabiendo esto hay muchas personas que no utilizan de forma apropiada las mascarillas, ya sea por desconocimiento o incomodidad. Muchas compañías y universidades han desarrollado diversas soluciones a esta problemática. Entre estas diversas soluciones podemos destacar, una aplicación *Web* que a través del reconocimiento facial y una cámara *Web*, detecta si la persona que está al frente de la cámara está usando una mascarilla o no [1]. Otra solución a esta problemática es un sistema de control que utiliza una cámara *web* y tarjeta *Raspberry Pi* reconoce si la persona tiene una mascarilla.

Este proyecto consiste en un sistema que tiene como base el reconocimiento facial con mascarillas usando una cámara que recibirá señales positivas o negativas de acuerdo a si la persona que este al frente de la cámara esté usando o no una mascarilla. La cámara se conecta a un dispositivo externo permitiendo el ingreso a ciertos lugares en específico, por ejemplo, un salón de clases, la cocina de un restaurante, entre otros.

En la figura 1, se muestra la funcionalidad del proyecto, en donde un usuario se coloca frente a la cámara, lo cual esta reconocerá el rostro del usuario y utilizando la aplicación de reconocimiento facial, compara con la información que cuenta y determinará si la persona que está frente a la cámara posee una mascarilla o la está utilizando de forma apropiada. Dependiendo del resultado permitiría el acceso a un determinado lugar.

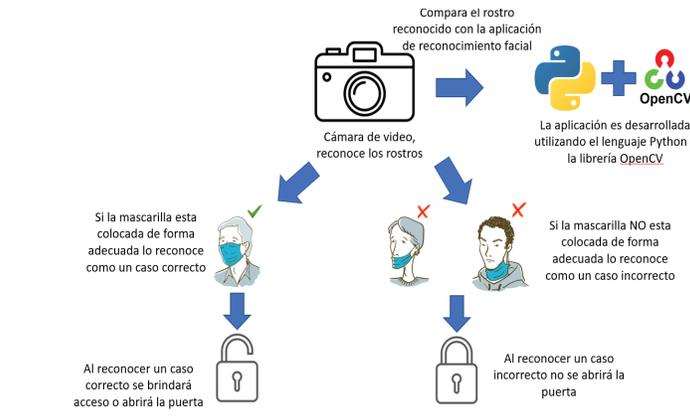


Figura 1. Diagrama del funcionamiento del proyecto.

Materiales y métodos (metodología)

Materiales

Los materiales que se utilizaran para realizar este proyecto:

Software:

- Visual Studio Code (Editor de Código)
- Lenguaje de programación Python
- Librería OpenCV
- Librería Numpy (Viene incluida en OpenCV)
- Librería MediaPipe

Hardware:

- Tarjeta Arduino MEGA 2560
- Luces Led
- Web Cam

Metodología

El proyecto consta de dos partes; la codificación para el reconocimiento facial y la confección del dispositivo.

Para la codificación se utilizó el editor de código *Visual Studio Code*, editor que es muy útil por su versatilidad y flexibilidad para utilizar diversos lenguajes con sus respectivas librerías.

El lenguaje de programación que se utiliza para este proyecto es *Python*, por dos razones principalmente; primero, por su sintaxis práctica y segundo, porque posee dos librerías que resultan muy útiles: *OpenCV* y *MediaPipe*.

OpenCV (Open Source Computer Vision) es una librería de visión por computadora iniciada por Intel en el año 1999. Esta librería multiplataforma se enfoca en el procesamiento de imágenes en tiempo real e implementaciones de los algoritmos de visión por computadora más recientes y de patente libre [3]. En el 2008 con la versión 2.3.1 por fin se incluyó esta librería en Python.

MediaPipe es una librería de *Python* que ofrece soluciones listas para usar, pero a su vez personalizables. El paquete *MediaPipe Python* está disponible en PyPI para Linux, macOS y Windows [4]. *MediaPipe* es un proyecto de *Google* que ofrece soluciones *Machine Learning* de código abierto, multiplataforma y personalizables para medios en vivo y de transmisión. En otras palabras, *MediaPipe* proporciona acceso a una amplia variedad de potentes modelos de aprendizaje automático creados teniendo en cuenta las limitaciones de hardware [5].

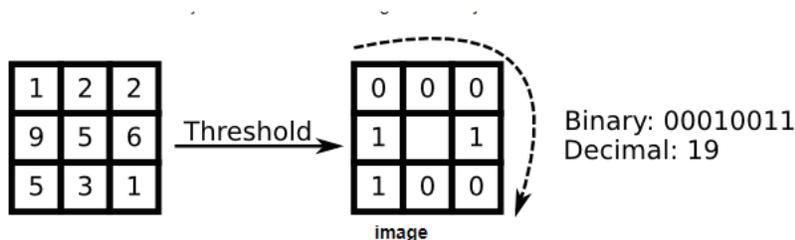


Figura 2. Ejemplo de cómo funciona el algoritmo LBPH con 8 píxeles. Fuente: [3]

De los métodos de reconocimiento que tiene *OpenCV* se utilizó *Local Binary Patterns Histograms (LBPH)*, el mismo funciona de manera adecuada. La idea básica de *LBPH* es resumir la estructura local de una imagen comparando cada píxel con su vecino. Si la intensidad del píxel central es mayor o igual al de su vecino lo indica con un 1 o 0 en caso contrario [3]. La figura 2 representa un diagrama de cómo funciona el algoritmo mencionado.



Figura 3. Tarjeta Arduino MEGA.

Para la confección del dispositivo se utilizó la webcam incorporada en una laptop y a su vez la computadora está conectada a una tarjeta Arduino que está vinculado con el código Python. La tarjeta está conectada a un Protoboard que tendrá un LED verde y un LED rojo. Cuando la

cámara detecte el rostro de una persona encenderá los LEDs, dependiendo de 2 condiciones encenderá un LED o el otro; si la persona tiene mascarilla puesta encenderá el LED de color verde, pero si la persona no tiene una mascarilla puesta encenderá el LED de color rojo. A continuación, en la figura 3 se muestra la tarjeta Arduino utilizada para la elaboración del prototipo.

Resultados

Al terminar lo anterior se comenzó la parte de prueba, donde se seleccionaron distintas imágenes y videos con el objetivo de entrenar y probar esta inteligencia artificial (IA), llegando a la conclusión de que sería más eficiente entrenar el programa con caras personas diferentes y a su vez distintos tipos de mascarillas variando también la forma de colocación y detalles como los colores. Después de un periodo de 2 semanas se logró concretar el objetivo planteado el cual fue que la IA lograra acertar en más del 70% de los casos. De modo que lo siguiente a realizar fue un prototipo funcional.

El cuadro 1 muestra los resultados de las 14 personas que fueron puestas a prueba la información fue procesada y almacenada en *Excel*.

Cuadro 1. Resultados de las pruebas realizadas a diversas personas.

Resultados				
No. de Personas puestas a prueba		Aciertos	Fallas	No identificados
14		9	3	2
% de error	17.29			
% de acierto	64.285			

Para realizar mejoras al tiempo de reconocimiento y una precisión más alta es necesario obtener una gran cantidad de videos de personas con el objetivo de que el entrenamiento sea más eficiente, también se pueden variar los rangos de detección, esto para que se minimice lo más posible el porcentaje de personas no identificadas.

Con la finalidad de pulir más los resultados se realizaron las siguientes pruebas mostradas en el cuadro 2, probando el funcionamiento con 10 personas.

Cuadro 2. Resultados de las pruebas realizadas solo bajándose un poco la mascarilla.

Resultados				
Bajándose la mascarilla				
No de Personas puestas a prueba		Aciertos	Fallas	No identificados
10		6	3	1
% de error	40			
% de acierto	60			

“Bajándose la mascarilla” es donde el programa tiene más dificultades con la detección por la forma en que funciona la librería, ya que la librería utiliza unos puntos de referencia para realizar reconocimiento facial (ejemplo, en los ojos, en la nariz, en los labios y en las orejas) y los puntos

de referencia entran en conflicto especialmente los que se encuentran en las orejas con el resto de los puntos, debido a que según los puntos de las ojeras aún se tiene la mascarilla puesta pero los otros puntos dicen lo contrario.

El cuadro 3 muestra los resultados cuando las personas puestas a prueba en un comienzo llevan mascarilla y luego se la quitaban.

Cuadro 3. Resultados de las pruebas quitándose la mascarilla del todo.

Resultados				
Quitándose la mascarilla				
No de Personas puestas a prueba		Aciertos	Fallas	No identificados
10		7	2	1
% de error	30			
% de acierto	70			

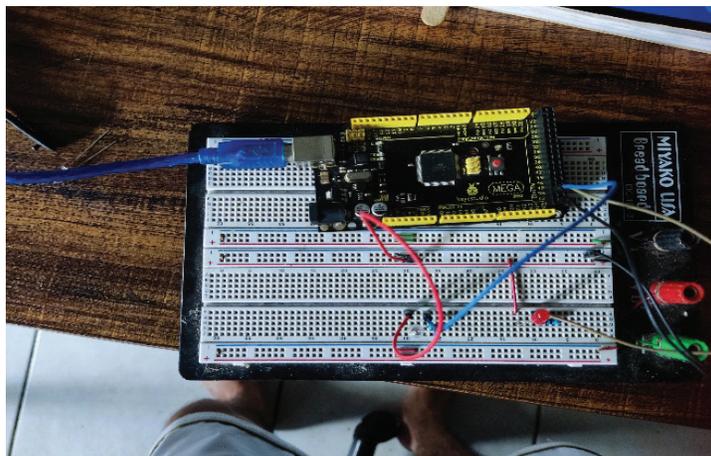


Figura 4. Foto del dispositivo armado.

Cabe destacar que cuando se creó y se implementó el dispositivo de *Arduino* se pensó que iba a tener conflictos cuando se acercaran dos personas con casos opuestos, es decir una con mascarilla y la otra sin mascarilla, pero el programa lo maneja moderadamente bien, basándose en el rostro que tenga más presencia en la cámara para encender el *LED* que corresponde. En la figura 4 se muestra el dispositivo ya terminado.

En la Figura 5 se puede ver cuando se probó la aplicación solamente el código, mientras que en la Figura 6 se probó ya con el dispositivo de *Arduino* implementado.

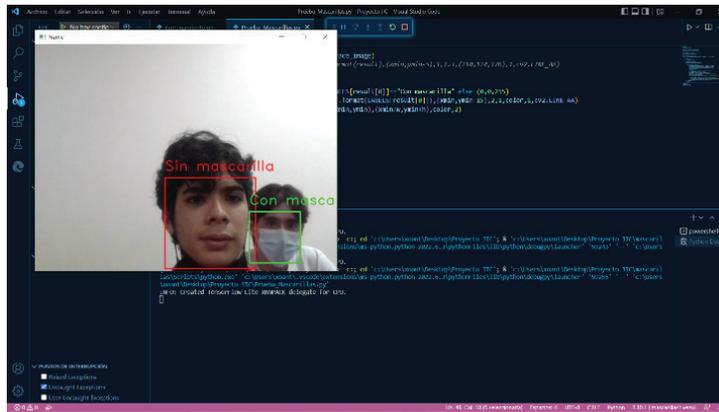


Figura 5. Pruebas realizadas solamente con el código.

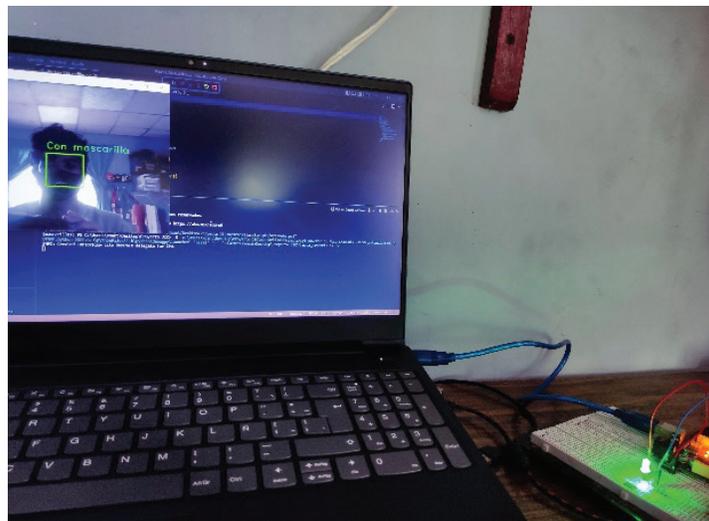


Figura 6. Pruebas realizadas con el dispositivo en Arduino.

Conclusiones

Terminado este proyecto se aprendió como implementar el reconocimiento de imágenes en proyectos prototipos usando Arduino, es importante también tener en cuenta que para desarrollar aplicaciones de Inteligencia Artificial esta debe ser entrenada previamente como lo fue en este caso, ya que se utilizaron diferentes videos de personas para tener una amplia base de datos y que así la IA sea más eficiente.

Dentro de las mejoras previstas para el futuro se tiene: Llevar a cabo un mejor control de reconocimiento a la hora que dos personas se paren al frente de la cámara, implementar la capacidad para que el sistema puede reconocer e indicar que tipo que mascarilla lleva la persona que se encuentre al frente de la cámara y poner en práctica un sistema que cuando reconozca un tipo de mascarilla permitida, este active un sistema de acceso para la persona.



Agradecimientos

A la Secretaria Nacional de Ciencia, Tecnología e Innovación de Panamá, por el financiamiento parcial para el desarrollo del proyecto. La Dra. Lilia Muñoz y el Dr. Vladimir Villarreal son miembros del Sistema Nacional de Investigación de la SENACYT.

Referencias

- [1] H. T. P. Donadon, «Aplicación de inteligencia artificial para monitorear,» *Revista Científica General José María Córdova*, vol. 19, n° 33, 2021.
- [2] V. d. I. Á. U. Toalombo, «SISTEMA DE CONTROL DE ACCESO POR MEDIO DE RECONOCIMIENTO,» Ambato – Ecuador, 2022.
- [3] W. Garage, «Face Recognition with OpenCV,» Google, [En línea]. Available: https://docs.opencv.org/4.2.0/d60/tutorial_face_main.html. [Último acceso: 24 mayo 2022].
- [4] G. LLC, «MediaPipe in Python,» 2020. [En línea]. Available: https://google.github.io/mediapipe/getting_started/python.html#:~:text=MediaPipe%20offers%20ready%2Dto%2Duse,%26%26%20source%20mp_env%2Fbin%2Factivate. [Último acceso: 28 Mayo 2022].
- [5] R. O'Connor, «AssemblyAI,» 7 Abril 2022. [En línea]. Available: <https://www.assemblyai.com/blog/mediapipe-for-dummies/>. [Último acceso: 28 Mayo 2022].



DETECCIÓN DE USO DE MASCARILLAS MEDICAS MEDIANTE RECONOCIMIENTO FACIAL

Guerra, Nikolai; Teira, Xoan; Castillo Jahir

Facultad de Ingeniería de Sistemas Computacionales
 Universidad Tecnológica de Panamá



RESUMEN

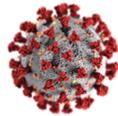
- Este proyecto se realizó como respuesta a algunas exigencias actuales las cuales dictan que el uso de mascarillas es obligatorio dentro de espacios cerrados como Colegios y Universidades.
- El objetivo principal de este proyecto es realizar un prototipo que pueda detectar si una persona está o no utilizando una mascarilla.
- Este proyecto de Inteligencia Artificial fue entrenado con diversos videos de personas mediante el reconocimiento de patrones detectó a una persona sin mascarilla, emitiendo una luz generada por un led roja y la generación de un sonido de alerta; si la persona cuenta con una mascarilla, solo se encenderá el led verde.

INTRODUCCION

El uso de mascarillas se ha vuelto cotidiano con la llegada de la pandemia del Covid-19 y el uso correcto es una de las principales medidas de seguridad. Muchas compañías y universidades han desarrollado diversas soluciones a esta problemática. Entre estas diversas soluciones podemos destacar, una aplicación Web que a través del reconocimiento facial y una cámara Web, detecta si la persona que está al frente de la cámara está usando una mascarilla o no. Otra solución a esta problemática es un sistema de control que utiliza una cámara web y tarjeta Raspberry Pi reconoce si la persona tiene una mascarilla.

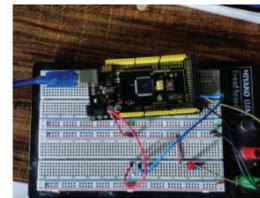
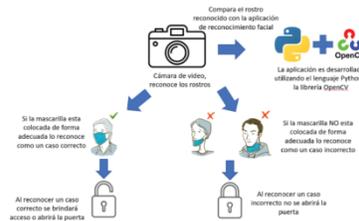
MATERIALES

- Software:**
- Visual Studio Code (Editor de Código)
 - Lenguaje de programación Python
 - Librería OpenCV
 - Librería Numpy (Viene incluida en OpenCV)
 - Librería MediaPipe
- Hardware:**
- Tarjeta Arduino MEGA 2560
 - Luces Led
 - Web Cam



METODOLOGIA

El proyecto a realizar consta de dos partes; la codificación para el reconocimiento facial y la confección del dispositivo.
 Para la codificación utilizamos el editor de código Visual Studio Code, editor que es muy útil por su versatilidad y flexibilidad para utilizar diversos lenguajes con sus respectivas librerías.

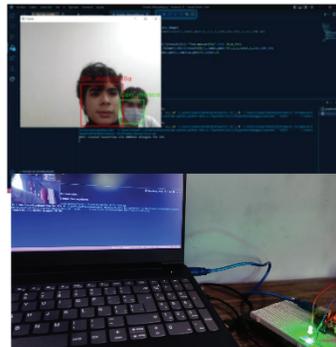


RESULTADOS

Después de un periodo de 2 semanas se logró concretar el objetivo planteado el cual fue que la IA lograra acertar en más del 70% de los casos. De modo que lo siguiente a realizar fue un prototipo funcional.

Resultados			
No. de Personas puestas a prueba	Aciertos	Fallas	No identificados
14	9	3	2

% de error 17.29
 % de acierto 64.285



CONCLUSION:

Terminado este proyecto se aprendió como implementar el reconocimiento de imágenes en proyectos prototipos usando Arduino, es importante también tener en cuenta que para desarrollar una inteligencia artificial esta debe ser entrenada previamente como lo fue en este caso, ya que se utilizaron diferentes videos de personas para tener una amplia base de datos y que así la IA sea más eficiente.