

# DEVELOPMENT OF AN ARTIFICIAL VISION ALGORITHM FOR THE RECOGNITION OF THE CORRECT USE OF EPP

Ryan Abraham León León<sup>1</sup>, Jessica Ayme León Montero<sup>2</sup>; <sup>1</sup>Universidad Privada Del Norte, Perú, ryan.leon@upn.edu.pe<sup>1</sup>, N00099438@upn.pe<sup>2</sup>

*Abstract: In this study, details the creation of a computer vision algorithm to verify the proper use of personal protective equipment (EPP). Using convolutional neural networks (CNN) together with tools such as Python 3.12 and YOLO V8, the system has been developed to identify in real time whether a person is wearing a mask and a headgear correctly. The accuracy of the system has been high thanks to its thorough training and validation, reaching a 96.57% validation rate. This approach is crucial to ensure that industries comply with Good Manufacturing Practices (GMP), thus guaranteeing consumer health and product quality. The development of the program included the installation and use of libraries such as OpenCV, Roboflow, MTCNN, Matplotlib, Imutils, Numpy and OS, which facilitated the detection of faces and EPPs, with training based on a database of more than 1500 labeled images.*

*Keywords: Computer Vision Algorithm, Python 3.12, YOLO V8, Convolutional Neural Networks (CNN), EPP.*

Digital Object Identifier (DOI):

ISBN:

ISSN:

# DESARROLLO DE UN ALGORITMO DE VISIÓN ARTIFICIAL PARA EL RECONOCIMIENTO DEL CORRECTO USO DE EPPS

Ryan Abraham León León<sup>1</sup>, Jessica Ayme León Montero<sup>2</sup>; <sup>1</sup>Universidad Privada Del Norte, Perú, ryan.leon@upn.edu.pe<sup>1</sup>, N00099438@upn.pe<sup>2</sup>

**Resumen:** *En este estudio, se detalla la creación de un algoritmo de visión artificial para verificar el uso adecuado de equipos de protección personal (EPP). Utilizando redes neuronales convolucionales (CNN) junto con herramientas como Python 3.12 y YOLO V8, el sistema ha sido desarrollado para identificar en tiempo real si una persona usa correctamente una mascarilla y una toca. La precisión del sistema ha sido elevada gracias a su entrenamiento y validación exhaustiva, alcanzando un 96.57% de validación. Este enfoque es crucial para asegurar que las industrias cumplan con las Buenas Prácticas de Manufactura (BPM), garantizando así la salud de los consumidores y la calidad de los productos. El desarrollo del programa incluyó la instalación y uso de librerías como OpenCV, Roboflow, MTCNN, Matplotlib, Imutils, Numpy y OS, las cuales facilitaron la detección de rostros y EPPs, con un entrenamiento basado en una base de datos de más de 1500 imágenes etiquetadas.*

**Palabras Claves:** *Algoritmo de Visión artificial, Python 3.12, YOLO V8, Redes neuronales convolucionales (CNN), EPP.*

## I. INTRODUCCIÓN

En las organizaciones manufactureras, los equipos de protección personal (EPP) desempeñan un rol crucial como la última barrera de defensa para asegurar la inocuidad de los productos y proteger a los empleados de riesgos específicos, reduciendo así la posibilidad de transmitir enfermedades. Con la pandemia, el uso de ciertos EPP, como las mascarillas, ha cobrado mayor relevancia, a pesar de que su uso en el entorno laboral no siempre fue prioritario. Actualmente, diversas empresas han integrado estos equipos como parte de su gestión de seguridad y salud ocupacional para garantizar tanto la seguridad de los productos como la salud de los trabajadores. En particular, el uso de tocas y mascarillas es fundamental para la producción de bienes que no pongan en riesgo la salud de los consumidores y para prevenir las enfermedades transmitidas por alimentos (ETA). El desarrollo de sistemas para la detección del uso correcto de mascarillas ha sido objeto de diversas investigaciones. En el estudio “Modelo para detectar el uso correcto de mascarillas en tiempo real utilizando redes neuronales convolucionales” [1], se diseñó un modelo utilizando TensorFlow basado en MobileNetV2, alcanzando una precisión del 94% con 25 interacciones, precisión del 94.67%, exhaustividad del 94.33% y un rendimiento del sistema del 94.33%. De igual manera, en

el artículo “Detección del uso de mascarillas mediante Visión Artificial y Redes Neuronales frente al Covid-19” se logró una precisión del 71.64% al detectar el uso de mascarillas mediante un programa de visión artificial basado en redes neuronales [2]. Otro estudio, titulado “Sistema de visión artificial para el reconocimiento de la mascarilla facial bien puesta” [3], diseñó un sistema que identifica si una mascarilla está bien colocada, logrando una precisión del 90% y un rendimiento del 95%. Además, en el trabajo “Desarrollo de un sistema de detección de rostros con o sin mascarilla mediante el uso de software” [4], se creó un software que detecta si una persona lleva puesta una mascarilla, alcanzando una precisión total del 83.13%, con una validación del 80% cuando la mascarilla está bien puesta, y del 86.25% cuando está mal puesta o no se usa. Otro artículo [5] analizó el reconocimiento facial de personas utilizando mascarillas o gafas mediante deep learning, alcanzando un 71% de acierto. Finalmente, una investigación [6] reportó el desarrollo de una aplicación de visión artificial que verifica el uso de mascarillas y tocas en industrias alimenticias, obteniendo un porcentaje de validación del 93.43% para mascarillas y del 92.86% para tocas. Conceptos clave como visión artificial, Python, YOLO, y redes neuronales convolucionales son fundamentales en este contexto. La visión artificial, una rama de la inteligencia artificial, se encarga de la obtención, procesamiento y análisis de información a partir de imágenes digitales [7]. Python, un lenguaje de programación de alto nivel, es ideal para aplicaciones educativas y técnicas, siendo utilizado en este tipo de investigaciones [8]. YOLO (You Only Look Once) es un sistema de código abierto que detecta objetos en tiempo real mediante redes neuronales convolucionales (CNN) [9]. Las redes neuronales convolucionales son ampliamente empleadas para el reconocimiento y procesamiento de imágenes, caracterizándose por su precisión [10]. OpenCV, una herramienta de visión por computador es esencial en este ámbito, especialmente en tareas académicas e investigativas [11].

A nivel mundial, los EPP son fundamentales en industrias como la construcción, fabricación y transporte. Sin embargo, el 83% de los empleadores enfrenta dificultades para asegurar su uso adecuado por parte de los trabajadores, quienes muchas veces los perciben como incómodos o innecesarios [12]. En

España, entre enero y septiembre de 2021, se registraron 28,315 accidentes laborales en el sector de transporte y almacenamiento, un 4.24% más que en el mismo período del año anterior [13]. Las redes neuronales convolucionales han mejorado significativamente en su desempeño, superando su uso exclusivamente académico y siendo implementadas en diversas aplicaciones prácticas [14]. Estas redes imitan el córtex visual humano, detectando características en imágenes mediante el procesamiento de píxeles [15]. Estudios comparativos muestran que las redes residuales (ResNet) ofrecen mejores resultados que las CNN en la detección de mascarillas, con una F1 del 90.1% frente a una F1 del 77.3% obtenida con MobileNetV2 [16]. Es fundamental considerar tanto el nivel de protección necesario como la comodidad, basándose en un análisis de riesgos que evalúe la duración y frecuencia de la exposición, las condiciones laborales y los posibles daños para los trabajadores [17].

La presente investigación se justifica tanto en términos prácticos como teóricos. Es necesario fiscalizar el uso de tocas y mascarillas para asegurar el cumplimiento de las Buenas Prácticas de Manufactura (BPM) y proteger la salud de los consumidores. Teóricamente, la investigación permitirá aplicar herramientas de visión artificial, como Python y YOLO, para desarrollar un algoritmo capaz de reconocer el correcto uso de EPP en tiempo real. Los objetivos incluyen la creación de un sistema de detección facial mediante visión artificial, el entrenamiento de una red neuronal convolucional, y el desarrollo de una base de datos para almacenar los resultados obtenidos.

## II. METODOLOGÍA Y MATERIALES

Las redes neuronales artificiales se han desarrollado como una forma de emular ciertas capacidades humanas, como la memoria y la asociación de hechos [18]. Cada red neuronal está compuesta por múltiples vías de entrada que funcionan de manera análoga a las dendritas en las neuronas biológicas. La información transmitida a través de estas vías llega al soma, que actúa como una función integradora. Posteriormente, una función de activación procesa esta información para generar una salida [19]. Para este estudio, se realizó una evaluación de los diferentes tipos de redes neuronales para determinar cuál sería la más adecuada para nuestro objetivo. Se concluyó que las Redes Neuronales Convolucionales (CNN) serían las más apropiadas debido a su superior rendimiento y eficiencia en tareas como la interpretación de imágenes y el reconocimiento de objetos [20]. Las CNN son esenciales para lograr el objetivo principal de esta investigación, que es desarrollar un algoritmo de visión artificial para identificar el uso correcto de equipos de protección personal (EPPs), particularmente para verificar si una persona está utilizando una mascarilla de manera adecuada. En la actualidad, las redes neuronales convolucionales (CNN) son ampliamente utilizadas y se destacan como un tipo fundamental de modelos de aprendizaje profundo. Estas redes han demostrado ser soluciones efectivas

en diversas áreas de investigación, como la visión por computadora y las aplicaciones de detección remota [21]. La gran ventaja de las CNN en el procesamiento de imágenes radica en su invarianza a la traducción; es decir, pueden generar mapas de activación similares incluso cuando la imagen sufre ligeras modificaciones, gracias a que la convolución actúa como un detector de características independientemente de su ubicación [22]. El sistema basado en YOLOv8 es altamente adaptable a diversas industrias, como la manufactura, construcción, salud y alimentación, donde la detección en tiempo real del uso correcto de equipos de protección personal (EPP) es esencial para cumplir con normativas de seguridad. Su implementación es accesible, requiriendo únicamente cámaras estándar y hardware como procesadores Intel Core i7 con 16GB de RAM, lo que permite su integración en sistemas existentes mediante librerías como OpenCV y Numpy.

En este proyecto, se implementó la red neuronal YOLOv8, la versión más reciente de esta arquitectura, debido a su capacidad para clasificar objetos en tiempo real, diferenciando entre categorías específicas como mascarillas y tocas. Esto ofrece una ventaja significativa en términos de velocidad y precisión en la detección, comparado con versiones anteriores de YOLO [23]. Además, La elección correcta de la función de activación es crucial para mejorar el rendimiento y la eficiencia de una red neuronal. En arquitecturas con capas convolucionales secuenciales, es fundamental usar una función de activación no lineal [24]. En este contexto, la función Leaky ReLU (A) se emplea como una mejora de la ReLU clásica, resolviendo problemas como la desactivación permanente de algunas neuronas durante el entrenamiento debido a la actualización de pesos [25]. A continuación, se presenta la ecuación 1 y la función de activación correspondiente (Figura 1).

$$f(x) = \begin{cases} 0, & \text{for } x < 0 \\ a * x, & \text{for } x \geq 0 \end{cases} \quad (1)$$

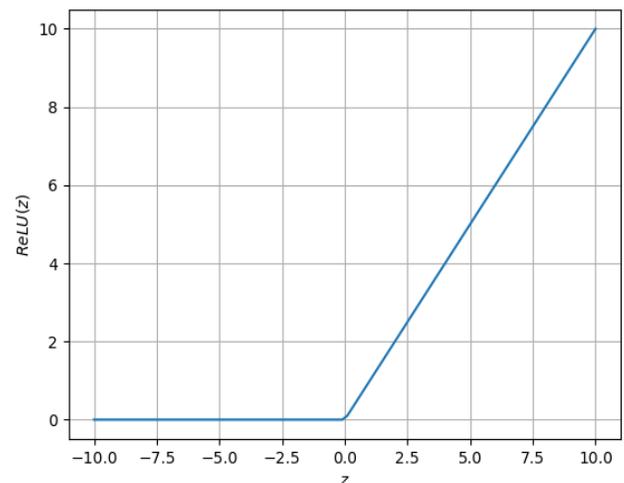


Figura 1 Gráfica de la Función ReLU (A).

Para el desarrollo del algoritmo, se utilizó como herramienta principal una laptop con capacidad de procesamiento adecuada, siendo fundamental para el funcionamiento del entorno de desarrollo Pycharm y el lenguaje Python, junto con diversas librerías que complementaron la codificación. Además, se empleó una cámara web de alta resolución, esencial para la captura y entrenamiento de imágenes dentro del sistema. Esta cámara fue responsable de capturar y enviar imágenes en tiempo real al algoritmo, permitiendo la identificación del uso correcto de mascarillas y tocas.

La Tabla I detalla los componentes y especificaciones necesarias para la codificación, así como los programas empleados en el desarrollo del algoritmo, el entrenamiento de imágenes y la detección de los equipos de protección personal.

TABLA I  
COMPONENTES PARA EL DESARROLLO DE LA CODIFICACIÓN.

Componente	Especificación
Procesador	11 th Gen Intel® Core™ i7-1165G7 @2.80 GHz
Capacidad RAM	16 GB
Sistema Operativo	Windows 11 Home
PyCharm	Versión 2024.1.1
Librerías	Opencv 4.9.0.80 Roboflow 1.1.34 Mtcnn 0.1.1 Matplotlib 3.9.0 Imutils 0.5.4 Numpy 1.26.4 OS

Por otro lado, se requirió la generación de una base de datos de imágenes para el entrenamiento del software. Para el desarrollo del algoritmo, es indispensable contar con una gran cantidad de imágenes correctamente etiquetadas, tanto para la etapa de entrenamiento como para la de validación. Es fundamental disponer de fotografías en las que se observe a las personas utilizando la mascarilla y/o toca de manera correcta, lo que permitirá al algoritmo identificar el uso adecuado. La distribución porcentual de las imágenes se detalla en la tabla II.

TABLA II  
DATASET SPLIT.

Base de datos	Detalle	%
TRAIN SET	1 476 imágenes	96%
VALID SET	46 imágenes	3%
TEST SET	16 imágenes	1%
Total	1 535	100%

#### A. PROCEDIMIENTO:

1. *Estructura de Visión Artificial – Capa 1:* El proceso de aprendizaje o entrenamiento de una red neuronal implica ajustar los parámetros. En esta etapa, que es la más crucial en

todo el desarrollo del sistema, se contó con la asistencia del programa PyCharm, utilizando el lenguaje de programación Python. Primero, se diseñó una base de datos con el propósito de entrenar el programa para la detección del uso correcto de mascarillas y tocas. Para ello, fue necesario etiquetar correctamente los objetos a detectar como se observa en la Figura 2.

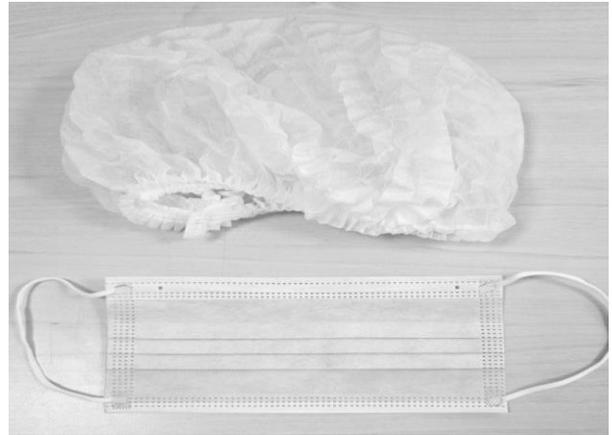


Figura 2 Objetos a detectar.

Para el continuo desarrollo del sistema, en la Tabla III, se detalla el correcto uso de la Toca y la Mascarilla que además de ser fundamental para garantizar la higiene y seguridad en industrias alimentarias, nos ayudará a la detección del uso correctos de estas.

TABLA III  
CORRECTO USO DE TOCA Y MASCARILLA.

TOCA	Mascarilla
<b>Cubrir completamente el cabello:</b> La toca debe cubrir todo el cabello, incluyendo las patillas, la nuca y cualquier mechón que pueda sobresalir.	<b>Cubrir nariz y boca:</b> La mascarilla debe cubrir completamente la nariz y la boca, asegurándose de que no haya espacios entre la mascarilla y la cara por donde puedan escapar gotículas.
<b>Ajuste adecuado:</b> La toca debe ajustarse firmemente. No debe moverse o caerse fácilmente durante el trabajo.	<b>Ajuste adecuado:</b> Debe ajustarse bien a los lados de la cara y debajo del mentón. Los elásticos o cintas deben ser cómodos pero firmes para evitar que la mascarilla se deslice.
<b>Limpieza:</b> Debe ser cambiada regularmente y siempre que esté sucia o dañada. <input type="checkbox"/>	<b>Evitar tocar la mascarilla:</b> Una vez colocada, se debe evitar tocarla con las manos, ya que esto puede transferir bacterias o virus a la superficie.
<b>Sin accesorios debajo:</b> No se deben llevar accesorios como aretes, gomas o pasadores debajo de la toca, ya que estos pueden causar que la toca no se ajuste bien o pueden caer en los alimentos.	<b>Cambio regular:</b> Estas deben ser cambiadas cada pocas horas, cuando estén visiblemente húmedas o sucias. Antes de ponerse o quitarse la mascarilla, se deben lavar las manos, además debe retirarse sólo los elásticos, evitando el contacto con la parte delantera.

Seguidamente el proceso de entrenamiento se inició con la entrada de las imágenes de la base de datos, donde se les asignó una etiqueta o identificador. Tras la finalización del entrenamiento, el conjunto de imágenes se utilizó para reconocer rostros con el "uso adecuado" e "inadecuado" de mascarillas y tocas, y para asignar las etiquetas correspondientes (Figura 3).

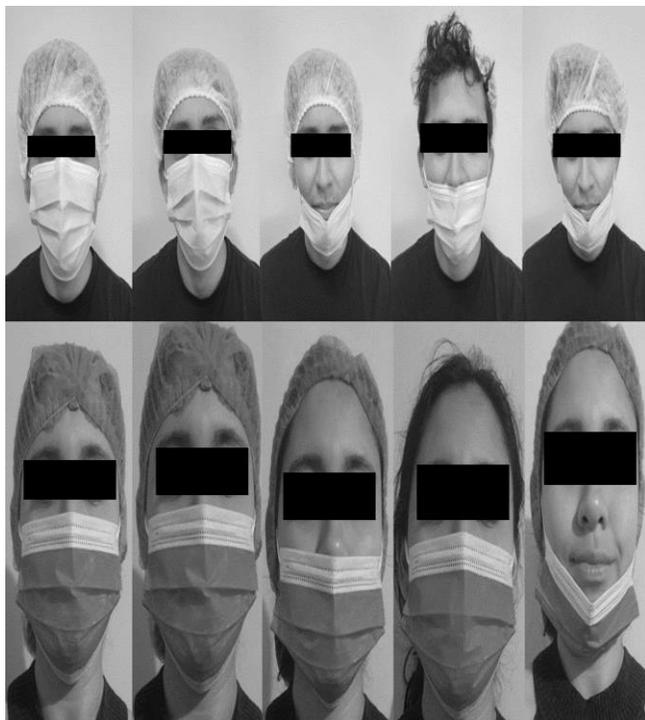


Figura 3 Captura de imágenes.

Posteriormente, para optimizar el reconocimiento, se procedió a redimensionar las imágenes de manera que se enfocaran en puntos específicos del rostro del sujeto, con el fin de minimizar errores en la detección. Se decidió capturar imágenes de 150 x 200 píxeles.

Se asignan los valores correspondientes a las coordenadas de un rectángulo, el cual será trazado en el centro de la pantalla. Estas coordenadas ( $x_1, y_1 = 190, 80$  y  $x_2, y_2 = 450, 398$ ) definen los límites de los objetos detectados y las áreas que los integran. Los píxeles están interconectados, lo que significa que cada pixel tiene cuatro vecinos en las direcciones horizontales y verticales. Este tipo de conexión es común en el procesamiento de imágenes para definir contornos o áreas. Las coordenadas se determinan mediante la ecuación 2, que se presenta a continuación.

$$(x + 1, y), (x - 1, y), (x, y + 1), (x, y - 1) \quad (2)$$

En la capa inicial del proceso, se describe el proceso de captura de imágenes de los EPPs (Equipos de Protección Personal), y la determinación de las coordenadas necesarias

para enfocar la detección. Este proceso de captura y enfoque es crucial para que el modelo pueda reconocer y procesar correctamente los objetos.

Además, la conexión de píxeles implica que el algoritmo no solo identifica objetos individuales, sino que también considera cómo están conectados los píxeles dentro de los objetos detectados, lo que podría mejorar la precisión del reconocimiento. En la tabla IV se detalla cómo el rectángulo es trazado con las coordenadas previamente asignadas, utilizando la función "cv2.rectangle" de OpenCV para dibujar un rectángulo azul alrededor de una región de la imagen. El fragmento "objeto = imAux [y1:y2, x1:x2]" extrae una subimagen contenida dentro del rectángulo, lo que resulta importante para el entrenamiento del modelo. Esta subimagen es utilizada como parte del proceso de aprendizaje, lo que sugiere que el algoritmo está diseñado para identificar objetos dentro de esa área específica de la imagen.

TABLA IV  
CÓDIGO MODELO DE ENTRENAMIENTO.

Códigos del algoritmo	Aspectos Seleccionados
$x_1, y_1 = 190, 80$ $x_2, y_2 = 450, 398$	Dimensiones de los píxeles conectados.
<code>cv2.rectangle(frame, (x1, y1), (x2,y2),(255,0,0),2)</code>	Establecimiento de una región con los puntos de coordenada.
<code>Objeto = imAux [y1:y2, x1:x2 ]</code>	Redimensionamiento de la región dibujada para entrenamiento.

Además, se llevó a cabo la segmentación que es una técnica que facilita la diferenciación entre los objetos presentes en el fondo de la imagen y los objetos a identificar. Esta etapa es importante para el reconocimiento de los objetos, ya que simplifica la escena. En el desarrollo del algoritmo se empleará la unión de regiones, donde cada píxel de las imágenes se tratará como una región. Este proceso comenzará con un conjunto inicial de píxeles, lo que hará que la región se expanda al incorporar los cuatro vecinos horizontales y verticales que comparten propiedades similares, como el color y el nivel de gris.

## B. ENTRENAMIENTO Y CODIFICACIÓN

1. *Estructura de Visión Artificial Capa 2:* En el primer bloque, Kernel D-Conv, se emplean filtros de 3x3 (K1, K2, K3) que realizan operaciones convolucionales sobre las entradas (I1, I2, I3), capturando patrones visuales como formas, texturas y bordes de tocas y mascarillas. En el segundo bloque, Kernel P-Conv, se utilizan filtros de 1x1 (K1, K2, K3) para combinar y refinar la información, facilitando la diferenciación entre diversas categorías de objetos. Este diseño permite una extracción y combinación eficiente de características, generando salidas (O1, O2, O3) que optimizan la precisión en la clasificación de la presencia o ausencia de tocas y mascarillas, apoyando tareas como el monitoreo de normativas y la identificación de objetos en seguridad (Figura 4).

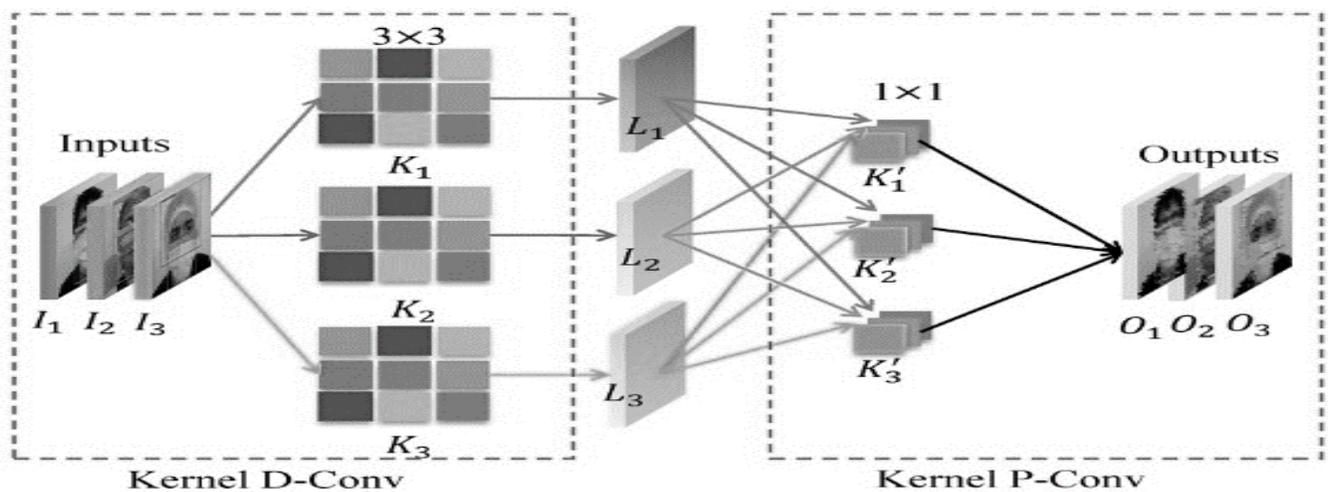


Figura 4 Kernel.

Los filtros convolucionales de 3x3 son ampliamente utilizados en el reconocimiento de imágenes debido a su capacidad para capturar características locales, como bordes o texturas, con una baja complejidad computacional [26]. Estas características son esenciales para identificar los objetos visuales en las imágenes de rostros

Las convoluciones 1x1 son comunes en arquitecturas de redes profundas (como ResNet o Inception) para combinar la información extraída en capas anteriores [27]. Este método permite al modelo hacer distinciones precisas entre diferentes categorías mejorando la capacidad del sistema para clasificar la presencia o ausencia de ciertos elementos (como el equipo de protección personal, EPP).

En aplicaciones de seguridad, es crucial que el modelo sea capaz de distinguir claramente entre personas que cumplen o no con las normativas de protección (como el uso de mascarillas) [28]. Este diseño de doble bloque convolucional optimiza la precisión en esta clasificación, mejorando el rendimiento del sistema.

El modelo implementado se entrena utilizando un conjunto de imágenes de rostros etiquetados, donde se asigna una etiqueta o identificador único a cada rostro. Al finalizar el entrenamiento, el conjunto de imágenes se emplea para identificar rostros no conocidos y asignar las etiquetas adecuadas. Este es un enfoque clásico de aprendizaje supervisado. El sistema aprende a reconocer objetos al asociar las características extraídas de las imágenes con las etiquetas asignadas a cada imagen de entrenamiento. Al final del proceso de entrenamiento, el modelo estará listo para clasificar correctamente nuevos datos.

De esta manera es que el sistema de visión artificial está diseñado para identificar objetos específicos como tocas y mascarillas mediante un enfoque de convoluciones profundas. Este utiliza una combinación de filtros 3x3 y 1x1 para extraer y refinar características visuales que optimizan la precisión del modelo en tareas de clasificación relacionadas con el monitoreo de seguridad y el cumplimiento de normativas, tal como se muestra en la Figura 5.



Figura. 5 Imágenes etiquetas para entrenamiento.

En la segunda capa del procedimiento se describe el código utilizado para el entrenamiento de las imágenes. El código empieza leyendo un directorio que contiene subdirectorios, con imágenes de rostros etiquetados. A medida que itera a través de subdirectorios y las imágenes de cada uno, se agregan las etiquetas y las imágenes de los rostros a listas separadas. Luego, se crea un objeto de reconocimiento facial, utilizando el algoritmo LBPH y se entrena utilizando las listas de imágenes y etiquetas. El método de entrenamiento se encarga de procesar las imágenes y generar un modelo capaz de reconocer rostros. A continuación, en la Tabla V se presenta la lectura del modelo de entrenamiento.

TABLE V  
CÓDIGO MODELO DE ENTRENAMIENTO.V

Códigos del algoritmo	Aspectos Seleccionados
<pre>!pip install roboflow --quiet from roboflow import Roboflow rf Roboflow(api_key="Owb7wL0INQDuAuA z9gth") project = rf.workspace("nicolai-hoirup- nielsen").project("cup-detection-v2") dataset project.version(3).download("yolov8") !yolo task=detect mode=train model=yolov8m.pt data={dataset.location}/data.yaml epochs=30 imgsz=640</pre>	Lectura del modelo entrenado.

El uso de subdirectorios etiquetados para almacenar las imágenes permite un sistema organizado y eficiente de clasificación, donde cada subdirectorio corresponde a un individuo o clase de objeto, lo que facilita la asociación entre imágenes y etiquetas. Además, el algoritmo LBPH es ampliamente utilizado para el reconocimiento facial debido a su capacidad para capturar patrones locales de los píxeles en una imagen y transformarlos en histogramas que representan las características del rostro. Este enfoque es especialmente útil en escenarios donde las condiciones de luz o ángulo de las imágenes pueden variar, ya que el LBPH es robusto frente a esas variaciones [29]. Y por último el método de entrenamiento basado en listas separadas de imágenes y etiquetas sigue el enfoque de aprendizaje supervisado, donde el modelo aprende a asociar las características visuales de las imágenes con las etiquetas correctas. Al finalizar, el modelo tiene la capacidad de reconocer y clasificar rostros no conocidos de manera eficiente. En conclusión, el código descrito en la Tabla V permite organizar las imágenes de rostros etiquetados en listas para entrenar un modelo de reconocimiento fácil utilizando el algoritmo LBPH y generar un modelo capaz de identificar el correcto uso de las mascarillas y tocas.

A continuación, se comenzó con la codificación, importando las bibliotecas necesarias para el desarrollo del algoritmo. Entre estas, se encuentra la biblioteca OpenCV, que facilita el manejo de imágenes en el entorno Python, una herramienta esencial para el procesamiento y análisis visual en este tipo de proyectos. También se utilizó la biblioteca NumPy, abreviada como np, que permite crear prototipos de algoritmos de manera eficiente y accesible, aprovechando su capacidad para manejar arreglos y realizar cálculos matemáticos de alto rendimiento. Finalmente, se empleó otra biblioteca fundamental para el proyecto la cula biblioteca MediaPipe, un framework multimodal y multiplataforma que integra machine learning, permitiendo el desarrollo de proyectos en dispositivos móviles, de escritorio o en la web. En la tabla VI se presenta la configuración del modelo de entrenamiento.

Tabla VI  
CÓDIGO DEL MODELO DE ENTRENAMIENTO.

Lectura del modelo	Descripción
<pre>from ultralytics import YOLO import cv2 # load yolov8 model Model = YOLO('C:/Users/Desktop/Mascarilla/ResultadosEntrenamiento/detect/train/weights/best.pt')</pre>	<p>Evento en el que se importa mejor modelo previamente entrenado para la correcta lectura en el programa.</p>

Después de la importación del modelo, se procede a su lectura, donde se ejecuta todo el algoritmo, dado que las imágenes han sido etiquetadas previamente y se entrenan adecuadamente con PyCharm. El propósito de este código es calcular y extraer toda la información almacenada en el

modelo, específicamente enfocada en la detección correcta del uso de mascarillas y tocas. Finalmente, se programó el proceso de captura de imágenes, especificando la cámara de vídeo encargada del funcionamiento del algoritmo, como se ilustra en la figura 6.



Figura 6 Detección en tiempo real.

2. Estructura de Visión Artificial Capa 3: En esta capa, se utiliza la técnica de escala de grises que permite clasificar las imágenes según su color de acuerdo con su parecido con otras imágenes de los grises de escala. El uso de la escala de grises en la tercera capa del sistema es fundamental para simplificar la imagen y resaltar las características más importantes, permitiendo al modelo hacer clasificaciones más precisas basadas en los niveles de gris. Esto es especialmente útil para la detección de objetos que presentan variaciones de color. (Figura 7).



Fig.7 Escala de Grises.

El programa realizar un mezclado de los tres canales RGB para convertir la imagen en una escala de grises, asignando los

siguientes porcentajes de ponderación a cada componente de color: verde 59%, rojo 30%, azul 11%. Estos porcentajes corresponden a la fórmula de luminancia que aparece en la ecuación (3), la cual describe matemáticamente cómo se ponderan los componentes de color para calcular la intensidad de gris en cada pixel.

$$(y = r \times 0.3 + g \times 0.59 + b \times 0.11) \quad (3)$$

Esta ecuación se aplica a cada pixel de la imagen RGB detectada, realizando la conversión a escala de grises. En la tercera capa del proceso, se incluyen los códigos necesarios para que el algoritmo ejecute esta conversión, así como las especificaciones de rendimiento de la imagen para optimizar la visualización y detección de los EPPs, como se indica en la tabla VII.

Códigos del algoritmo	Descripción
<code>gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)</code>	Aspectos seleccionados
<code>cv2.rectangle(frame, (x, y),(x + w, y + h),(0,255,0),2)</code> <code>cv2.putText(frame,'EPP', (x, y-10),2,0.7, (0,255,0),2, cv2.LINE_AA)</code>	Establecimiento de medidas para la visualización de los EPP.

3. *Estructura de la red neuronal:* Es importante destacar la estructura neuronal utilizada en este desarrollo ya que es fundamental para procesar y extraer características de las imágenes, lo que facilita la detección automática de EPPs. Basada en lo previamente descrito, la Figura 8 presenta la arquitectura de visión artificial de la red neuronal para facilitar una mejor comprensión del proceso.

TABLA VII  
CÓDIGO DE APLICACIÓN DE ESCALA DE GRIS Y REDIMENSIONAMIENTO

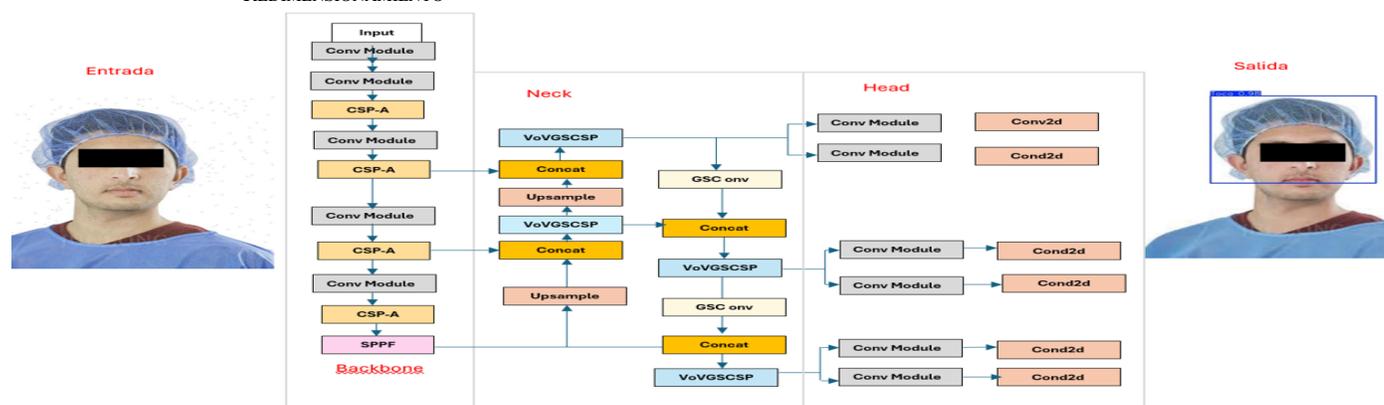


Figura 8 Arquitectura de la Red Convolutiva YoloV8.

### III RESULTADOS

#### A. Entrenamiento por épocas

Para el entrenamiento de este algoritmo, se procesó un conjunto de datos de 1535 imágenes a lo largo de 250 épocas. En cada época, el algoritmo recopiló información relevante que le permitió aumentar su precisión. Aunque al principio cometía numerosos errores, estos disminuyeron progresivamente a medida que avanzaba el entrenamiento, mejorando así sus resultados. En la Figura 9 observamos el

diagrama Box\_loss, este ayuda al modelo a aprender la posición y el tamaño correcto de los cuadros delimitadores alrededor de los objetos detectados. También está el diagrama Class\_Loss, que garantiza que el modelo identifique con mayor precisión la categoría del objeto detectado. Y por último, el diagrama DistributionFocal Loss, este en lugar de predecir directamente las coordenadas del cuadro, predice la distribución de posibles desplazamientos del cuadro.

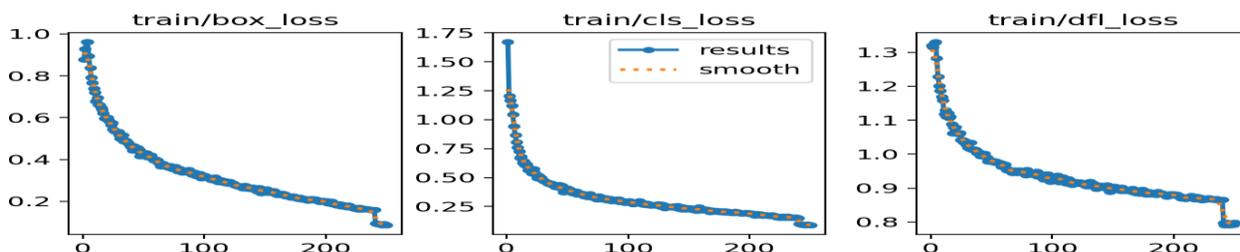


Figura 9 Gráficos del entrenamiento

### B. Precisión

Tras realizar las pruebas, se obtiene el gráfico mostrado en la Figura 10, el cual presenta un resultado positivo debido a su tendencia ascendente cercana a 1. Esto sugiere una precisión superior al 90% en el modelo.

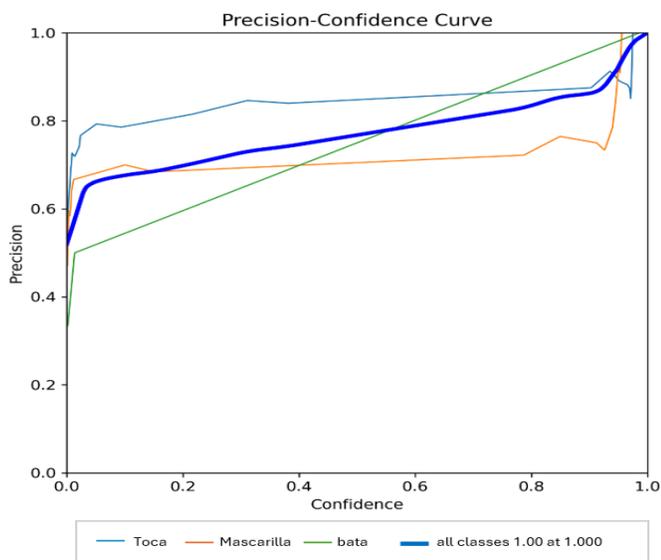


Fig. 10 Gráfico de precisión

Como se muestra en la Figura 11, la matriz de confusión ilustra las comparaciones entre las imágenes de prueba y las etiquetas reales. Los valores en la diagonal principal indican las detecciones correctas, como las mascarillas con un 88%, las tocas con un 72% y las batas con un 100%. Por otro lado, los valores fuera de esta diagonal reflejan errores, representando las imágenes que presentaron dificultades en la detección. Además, se consideró la detección de batas para reducir los falsos positivos, dado que existe una similitud entre los objetos.

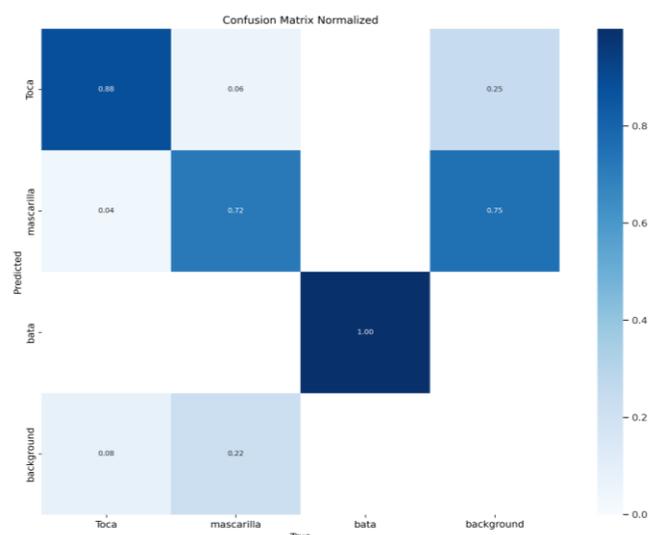


Fig. 11 Matriz de confusión expresado en porcentaje.

### C. Validación

Se recopilaron los datos de las pruebas de validación en las siguientes tablas. Se llevaron a cabo 70 pruebas, alternando entre 35 realizadas con mascarilla y/o toca y 35 sin mascarilla y/o toca. Los resultados obtenidos se detallan en las Tablas VIII, IX y X, donde se muestra la frecuencia de respuestas correctas e incorrectas del sistema, junto con el porcentaje de validación correspondiente.

La primera prueba de validación consistió en evaluar la precisión del reconocimiento del uso correcto o incorrecto de mascarillas. Los resultados se detallan en la Tabla VIII

TABLA VIII  
RESULTADOS DE LAS PRUEBAS DE VALIDACIÓN DE SUJETOS PARA EL CORRECTO USO DE MASCARILLAS

Sujetos	Con mascarilla (correcto)	Con mascarilla (Incorrecto)	Sin mascarilla (correcto)	Sin mascarilla (Incorrecto)	Total
1	35	0	32	3	70
2	33	2	33	2	70
3	34	1	34	1	70
4	35	0	35	0	70
5	35	0	33	2	70
Total	172	3	167	8	350
%	98.29%	1.71%	95.43%	2.23%	

La Tabla VIII indica que el sistema mostró un alto nivel de precisión en el reconocimiento del uso de mascarillas, con un 98.29% de aciertos cuando los sujetos llevaban mascarilla y un 95.43% de precisión cuando no llevaban. Los errores fueron mínimos, con un 1.71% y 2.23% de respuestas incorrectas, respectivamente. Estos resultados sugieren que el algoritmo de reconocimiento es altamente confiable para detectar la presencia de mascarillas, validando así su uso en entornos de seguridad donde es necesario cumplir con las normativas de protección.

Como segunda prueba de validación se probó el reconocimiento del uso correcto o incorrecto de tocas. Los resultados se presentan en la Tabla IX.

TABLE IX  
RESULTADOS DE PRUEBA DE VALIDACIÓN DE SUJETOS PARA EL CORRECTO USO DE TOCA.

Sujetos	Con mascarilla (correcto)	Con mascarilla (Incorrecto)	Sin mascarilla (correcto)	Sin mascarilla (Incorrecto)	Total
1	34	1	34	1	70
2	35	0	35	0	70
3	34	1	34	1	70
4	35	0	35	0	70
5	33	2	32	3	70
Total	171	4	170	5	350
%	97.71%	2.29%	97.14%	1.41%	

La tabla IX indica que el sistema fue igualmente eficiente en la detección del uso de tocas, con un 97.71% de precisión cuando lo sujetos llevaban tocas y un 97.14% de aciertos cuando no las llevaban. Por lo tanto, la baja tasa de errores refuerza la capacidad del algoritmo para identificar con precisión si una toca está presente o no, lo cual es crucial en el cumplimiento de normativas de protección en áreas específicas. Como tercera prueba de validación se evaluó la detección simultánea de mascarilla y toca. Los resultados se describen en la Tabla X.

TABLA X  
RESULTADOS DE PRUEBA DE VALIDACIÓN DE SUJETOS PARA EL CORRECTO USO DE MASCARILLA Y TOCA (EPPS).

Sujetos	Con mascarilla (correcto)	Con mascarilla (Incorrecto)	Sin mascarilla (correcto)	Sin mascarilla (Incorrecto)	Total
1	33	2	35	0	70
2	35	0	35	0	70
3	34	1	34	1	70
4	35	0	35	0	70
5	32	3	35	0	70
Total	169	6	174	1	350
%	96.57%	3.43%	99.43%	0.28%	

La Tabla X indica la detección de ambos EPPs de manera simultánea alcanzó una precisión del 96.57% cuando los sujetos los llevaban, y un 99.43% cuando no los utilizaban. Aunque el sistema mostró más errores en la detección conjunta, los niveles de precisión siguen siendo altos.

El modelo YOLOv8 fue entrenado utilizando un conjunto de datos de 1,535 imágenes, dividido en entrenamiento, validación y prueba. Se implementaron técnicas de preprocesamiento como redimensionamiento y segmentación para mejorar la precisión. Durante 250 épocas, el modelo ajustó sus parámetros mediante una función de activación Leaky ReLU, alcanzando una precisión promedio del 96.57% en la validación. El uso de métricas como la matriz de confusión permitió evaluar y ajustar el rendimiento, garantizando su efectividad en la detección de mascarillas y tocas en tiempo real.

#### DISCUSIÓN

En el artículo [30], se concluye que la detección del uso de implementos mediante visión artificial y redes neuronales logra un nivel de precisión del 71.64%. En comparación, en el presente estudio se alcanzó un porcentaje de más de 95%, lo que representa un nivel de precisión aceptable para su implementación con redes neuronales. Además, se señala que en el artículo mencionado se trabajó con un total de 877 imágenes, mientras que en este estudio se utilizaron 1535, con el objetivo de mejorar el aprendizaje en la detección. Este resultado también demuestra que el uso del programa Python

para la detección de EPPs es eficaz, permitiendo un control más eficiente de los sistemas de monitoreo en comparación con métodos manuales. Por otro lado, en el artículo [6], se obtuvo un porcentaje de validación del 95.43% para la detección del uso correcto de ambos implementos y un 92% cuando no se utilizan. En contraste, en este estudio, la validación de la detección del uso correcto de ambos EPPs alcanzó un 96.57% de precisión, y un 99.43% de precisión cuando no se utilizan correctamente, lo que indica un mejor desempeño en los resultados obtenidos.

Es importante destacar que los resultados de esta investigación presentan similitudes con ambos proyectos, ya que el sistema de detección de tocas y mascarillas se desarrolló utilizando Python y bibliotecas comunes como Numpy, Matplotlib, TensorFlow, y OpenCV. Este enfoque se considera beneficioso para las empresas, ya que permite un mayor control de riesgos relacionados con enfermedades y prácticas higiénicas, especialmente en industrias alimentarias, donde es crucial evitar errores por parte de los trabajadores para preservar la inocuidad alimentaria.

#### CONCLUSIONES

Se consiguió desarrollar un sistema de detección facial utilizando visión artificial y algoritmos de inteligencia artificial, implementado en PyCharm con el lenguaje de programación Python, y empleando diversas bibliotecas como OpenCV, Roboflow, MTCNN, Matplotlib, Imutils, Numpy y OS para completar la codificación. Se trabajó con una red neuronal convolucional, en la cual se realizó el entrenamiento; este proceso incluyó la descripción del código, la adición de etiquetas y la clasificación de las imágenes de los rostros en listas separadas. Posteriormente, el entrenamiento se llevó a cabo utilizando estas listas de imágenes y etiquetas, permitiendo al modelo procesar las imágenes y generar una estructura capaz de reconocer rostros. Además, se creó una base de datos para recopilar y almacenar los datos obtenidos, donde inicialmente se etiquetaron correctamente los objetos a detectar en un total de 1535 imágenes, divididas en Entrenamiento (1476 imágenes), Validación (43 imágenes) y Prueba (16 imágenes). Además, que el sistema YOLOv8, aunque muestra resultados positivos con una precisión del 96.57%, puede enfrentar dificultades en condiciones de baja iluminación, alta densidad de personas o cuando los objetos están parcialmente obstruidos, lo que genera falsos positivos o negativos. También puede confundirse con objetos similares a las mascarillas y tocas. Para mitigar estos problemas, se podrían usar cámaras infrarrojas y ajustar el preprocesamiento de imágenes para mejorar su precisión en escenarios complejos. La investigación concluye que el sistema desarrollado presenta una alta precisión en el reconocimiento del uso correcto de EPPs (mascarilla y toca), alcanzando un 96.57%, lo cual indica que la calidad del modelo es satisfactoria y que podría ser utilizado por diferentes empresas como parte de su gestión en seguridad y salud ocupacional.

## REFERENCIAS

- [1] CARRERA, Hugo Andrade; MAITA, Soraya Sinche; LASCANO, Pablo Hidalgo. Modelo para detectar el uso correcto de mascarillas en tiempo real utilizando redes neuronales convolucionales. *Revista de Investigación en Tecnologías de la Información: RITI*, 2021, vol. 9, no 17, p. 111-120. <https://riti.es/index.php/riti/article/view/78>
- [2] MARCHENA, Teodoro Geldres, et al. Detección del uso de mascarillas mediante Visión Artificial y Redes Neuronales frente al Covid-19. *INGnosis*, 2022, vol. 8, no 1, p. 01-09. [https://www.researchgate.net/publication/369029348\\_Deteccion\\_del\\_uso\\_de\\_mascarillas\\_mediante\\_Vision\\_Artificial\\_y\\_Red\\_Neuronales\\_frente\\_al\\_Covid-19H](https://www.researchgate.net/publication/369029348_Deteccion_del_uso_de_mascarillas_mediante_Vision_Artificial_y_Red_Neuronales_frente_al_Covid-19H). Simpson, *Dumb Robots*, 3<sup>rd</sup> ed., Springfield: UOS Press, 2004, pp.6-9.
- [3] QUINTERO, Trujillo, et al. Sistema de Visión Artificial para el Reconocimiento de la Mascarilla Facial Bien Puesta. 2023. [https://rinacional.tecnm.mx/bitstream/TecNM/5114/1/MC\\_Jose\\_Omar\\_de\\_Jesus\\_Trujillo\\_Quintero\\_2023.pdf](https://rinacional.tecnm.mx/bitstream/TecNM/5114/1/MC_Jose_Omar_de_Jesus_Trujillo_Quintero_2023.pdf)
- [4] LEÓN LEÓN, Ryan Abraham, et al. Development of a face detection system with or without a mask using software. [https://laccei.org/LACCEI2022-BocaRaton/full\\_papers/FP753.pdf](https://laccei.org/LACCEI2022-BocaRaton/full_papers/FP753.pdf) (LACCEI)
- [5] Mucha, J. E., Ortega, A. R., Salazar, L. M. R., Montero, M. M. A., Pahuacho, R. C. U., & Moreno, A. E. G. (2021). Aplicación del deep learning para el reconocimiento facial con la presencia de oclusiones en el contexto de la pandemia covid 2021. *Revista ECIPerú Volumen*, 18(1). DOI: <https://doi.org/10.33017/RevECIPeru2021.0002/>
- [6] LEÓN LEÓN, Ryan Abraham, et al. Artificial verification system on the use of safety elements. *LACCEI*, 2023, vol. 1, no 8. (LACCEI) [https://laccei.org/LACCEI2023-BuenosAires/papers/Contribution\\_552\\_a.pdf](https://laccei.org/LACCEI2023-BuenosAires/papers/Contribution_552_a.pdf)
- [7] GONZÁLEZ MEYBERG, Camilo Eduardo; ZARAMA GARZÓN, Daniel Fernando. Aplicación de sistemas de visión por computador (VIR/NIR) para la inspección visual de uchuvas. <https://core.ac.uk/download/pdf/71419539.pdf>
- [8] Challenger-Pérez, Ivett, Díaz-Ricardo, Yanet, Becerra-García Roberto Antonio. El lenguaje de programación Python. *Ciencias Holguín* [en línea]. 2014, XX(2), 1-13[fecha de Consulta 3 de junio de 2024]. ISSN: Disponible en: <https://www.redalyc.org/articulo.oa?id=181531232001>
- [9] BARBUDO GONZÁLEZ, Sebastián. Comparación de algoritmos basados en inteligencia artificial para la detección del uso del tapabocas. 2021. <https://repositorio.escuelaing.edu.co/bitstream/handle/001/1978/Barbudo%20Gonz%C3%A1lez%2c%20Sebasti%C3%A1n.-2021.pdf?sequence=1&isAllowed=y>
- [10] PAULLÁN PUNGUILL, Alexandra Elizabeth. Implementación de un prototipo de detección de mascarilla y medición de temperatura para el personal que ingresa a la empresa PAUFIT, usando técnicas de visión artificial. 2021. <http://dspace.esoch.edu.ec/handle/123456789/20475>
- [11] ARÉVALO, Vicente; GONZÁLEZ, J.; AMBROSIO, Gregorio. La librería de visión artificial opencv. aplicación a la docencia e investigación. *Base Informática*, 2004, vol. 40, p. 61-66. <http://mapir.isa.uma.es/varevalo/drafts/arevalo2004lva1.pdf>
- [12] ALLEN LAW GROUP. El problema del EPP: un estudio culpa a los trabajadores por no usar el equipo de protección personal en el trabajo. [En línea][citado el 25 de abril del 2024] Disponible en <https://www.kenallenlaw.com/es/2022/10/the-ppe-problem-study-blames-workers-for-failing-to-use-personal-protective-equipment-on-the-job/>
- [13] MASCONTAINER. Accidentes laborales al alza en transporte y logística. [En línea][citado el 25 de abril del 2024] Disponible en <https://www.mascontainer.com/accidentes-laborales-al-alza-transporte-y-logistica/>
- [14] CIFUENTES, Adriana, et al. Desarrollo de una red neuronal convolucional para reconocer patrones en imágenes. *Investigación y desarrollo en TIC*, 2019, vol. 10, no 2, p. 7-17. <https://revistas.unisimon.edu.co/index.php/identica/article/download/4007/4359>
- [15] ARTOLA MORENO, Álvaro. Clasificación de imágenes usando redes neuronales convolucionales en Python. 2019. <https://idus.us.es/bitstream/handle/11441/89506/TFG-2402-ARTOLA.pdf?sequence=1>
- [16] BARBUDO GONZÁLEZ, Sebastián. Comparación de algoritmos basados en inteligencia artificial para la detección del uso del tapabocas. 2021. <https://repositorio.escuelaing.edu.co/bitstream/handle/001/1978/Barbudo%20Gonz%C3%A1lez%2c%20Sebasti%C3%A1n.-2021.pdf?sequence=1&isAllowed=y>
- [17] Pérez, S. (2012). Elementos de protección personal. [https://www.famaf.unc.edu.ar/documents/498/Manual\\_de\\_elementos\\_de\\_proteccion%20C3%B3n\\_personal.pdf](https://www.famaf.unc.edu.ar/documents/498/Manual_de_elementos_de_proteccion%20C3%B3n_personal.pdf)
- [18] TICH, Damián Jorge. *Redes Neuronales: Conceptos básicos y aplicaciones*. Universidad Tecnológica Nacional, México, 2001, vol. 41, p. 12-16. [https://www.frro.utn.edu.ar/repositorio/catedras/quimica/5\\_anio/orientador1/monograias/match-redesneuronales.pdf](https://www.frro.utn.edu.ar/repositorio/catedras/quimica/5_anio/orientador1/monograias/match-redesneuronales.pdf)
- [19] DEL BARRIO, A. Pérez, et al. Inteligencia artificial en Radiología: introducción a los conceptos más importantes. *Radiología*, 2022, vol. 64, no 3, p. 228-236. <https://doi.org/10.1016/j.rx.2022.03.003> (SCOPUS)
- [20] GU, Jiuxiang, et al. Recent advances in convolutional neural networks. *Pattern recognition*, 2018, vol. 77, p. 354-377. <https://doi.org/10.1016/j.patcog.2017.10.013> (SCIENCE)
- [21] GÓMEZ PUJANTE, Begoña. *Redes Convolucionales. Aplicación a la clasificación de imágenes médicas*. 2023. <https://hdl.handle.net/11000/30233>
- [22] LI, Jun, et al. Transforming medical imaging with Transformers? A comparative review of key properties, current progresses, and future perspectives. *Medical image analysis*, 2023, vol. 85, p. 102762. <https://doi.org/10.1016/j.media.2023.102762> (SCIENCE)
- [23] RODRÍGUEZ, Albert I.; BUITRAGO, Xiomara D. How to choose an activation function for deep learning. *Tekhnè*, 2022, vol. 19, no 1, p. 23-32. <https://geox.udistrital.edu.co/index.php/tekhne/article/view/20337>
- [24] MORENO, Andrea Navarro. Introducción a las Redes Neuronales aplicadas al Aprendizaje Supervisado. <https://scholar.archive.org/work/wnhw7sw3cjr7dnnhiqbq5i74/access/w/awayback/https://s3-eu-west-1.amazonaws.com/pfigshare-ufiles/20834220/265A.NavarroM.InteligenciaArtificial.pdf>
- [25] VILAGRAN SOLSONA, Albert. Facial Expression Detection using Convolutional Neural Networks. 2018. Tesis de Licenciatura. Universitat Politècnica de Catalunya. <http://hdl.handle.net/2117/129218>
- [26] MUÑOZ SÁNCHEZ, Francisco, Estudio del reconocimiento de emociones en rostros mediante inteligencia artificial y el impacto del color de piel. Cartagena, Octubre 2023 <https://repositorio.upct.es/server/api/core/bitstreams/01b4b91c-d387-4c87-92dc-b229e1b29462/content>
- [27] TOMAS MARTÍN, Alberto; ROBLES FORCADA, Victor. Investigación de Detección de Bordes a Radiografías con Deep Learning, Madrid mayo 2023 [https://oa.upm.es/74981/1/TFG\\_ALBERTO\\_TOMAS\\_MARTIN.pdf](https://oa.upm.es/74981/1/TFG_ALBERTO_TOMAS_MARTIN.pdf)
- [28] BENITES, Jorge Bryhan., Desarrollo de un clasificador de uso correcto de mascarillas usando redes neuronales convolucionales. <https://dspace.unitru.edu.pe/server/api/core/bitstreams/5823e296-6535-4031-9e76-e4b971c36b08/content>.
- [29] BUDIMAN André, et al. A Student attendance with face recognition (LBPH OR CNN): Systematic literature review, 2023. [https://www.researchgate.net/profile/Said-Achmad/publication/366998544\\_Student\\_attendance\\_with\\_face\\_recognition\\_LBPH\\_or\\_CNN\\_Systematic\\_literature\\_review/links/63c77f69d9f5967c2e55958/Student-attendance-with-face-recognition-LBPH-or-CNN-Systematic-literature-review.pdf](https://www.researchgate.net/profile/Said-Achmad/publication/366998544_Student_attendance_with_face_recognition_LBPH_or_CNN_Systematic_literature_review/links/63c77f69d9f5967c2e55958/Student-attendance-with-face-recognition-LBPH-or-CNN-Systematic-literature-review.pdf)
- [30] MARCHENA, Teodoro Geldres, et al. Detección del uso de mascarillas mediante Visión Artificial y Redes Neuronales frente al Covid-19. *INGnosis*, 2022, vol. 8, no 1, p. 01-09. [https://www.researchgate.net/publication/369029348\\_Deteccion\\_del\\_uso\\_de\\_mascarillas\\_mediante\\_Vision\\_Artificial\\_y\\_Red\\_Neuronales\\_frente\\_al\\_Covid-19H](https://www.researchgate.net/publication/369029348_Deteccion_del_uso_de_mascarillas_mediante_Vision_Artificial_y_Red_Neuronales_frente_al_Covid-19H). Simpson, *Dumb Robots*, 3<sup>rd</sup> ed., Springfield: UOS Press, 2004, pp.6-9.