Revista Innovación y Software
Vol. 4, No. 1, Mes Marzo-Agosto, 2023
ISSN: 2708-0935
Pág. 24-36
https://revistas.ulasalle.edu.pe/innosoft

Tipo de artículo: Artículos originales
Temática: Inteligencia artificial
Recibido: 22/09/2022 | Aceptado: 05/11/2022 | Publicado: 30/03/2023

Identificadores persistentes:
ARK: ark:/42411/s11/a78
PURL: 42411/s11/a78

Sistema de reconocimiento facial para el control de accesos mediante Inteligencia Artificial

Facial recognition system for access control through Artificial Intelligence

Jean Elias Manuel Reyes Campos [1][0000-0001-8635-4560]*, Christian Stephano Castañeda Rodríguez [2][0000-0003-1409-7870], Luis Daniel Alva Luján [3][0000-0003-1587-6366], Alberto Carlos Mendoza de los Santos [4][0000-0002-0469-915X]

[1] Universidad Nacional de Trujillo. Dirección postal. jereyesc@unitru.edu.pe
[1] Universidad Nacional de Trujillo. Dirección postal. ccastaneda@unitru.edu.pe
[1] Universidad Nacional de Trujillo. Dirección postal. ldalval@unitru.edu.pe
[1] Universidad Nacional de Trujillo. Dirección postal. amendozad@unitru.edu.pe

* Autor para correspondencia: jereyesc@unitru.edu.pe

Resumen

El presente artículo tiene como objetivo principal el desarrollo de un sistema que permita el reconocimiento facial de una persona para el control de accesos mediante Inteligencia Artificial. Para el desarrollo del sistema se tuvo como algoritmo Redes Neuronales Convolucionales, el cual es un modelo de reconocimiento. Así mismo se utilizó el lenguaje de programación Python y las librerías siguientes como Numpy, Os, OpenCV e Imutils para su implementación. Los resultados obtenidos según el acierto y utilizando un dataset de 450 imágenes por individuo son de un 88% aproximadamente en cuanto la predicción por persona, concluyendo que el sistema de reconocimiento es eficaz y tiene mayor eficiencia incrementando el tamaño de datasets generados por individuos.

Palabras clave: Control de acceso, Inteligencia Artificial, Redes Neuronales Convolucionales.

Abstract

The main objective of this article is the development of a system that allows the facial recognition of a person for access control through Artificial Intelligence. For the development of the system, the Convolutional Neural Networks algorithm was used, which is a recognition model. Likewise, the Python programming language and the following libraries such as Numpy, Os, OpenCV and Imutils were used for its implementation. The results obtained according to the hit and using a dataset of 4500 images are approximately 88% in terms of the prediction per person, concluding that the recognition system is effective and has greater efficiency by increasing the size of datasets generated by individuals.

Revista Innovación y Software
Vol. 4, No. 1, Mes Marzo-Agosto, 2023
ISSN: 2708-0935
Pág. 24-36
https://revistas.ulasalle.edu.pe/innosoft

Keywords: *Access Control, Artificial Intelligence, Convolutional Neural Networks.*

Introducción

Actualmente la relación entre computadora y humano va reduciendo brechas, lo cual genera consigo una gran necesidad en implementación de seguridad informática. La seguridad informática es considerada un tema de gestión alineado a estándares y buenas prácticas [13], por lo cual requiere de un sistema de autenticación para restringir el acceso de los usuarios a cierta información almacenada en computadores.

Dentro de los diferentes tipos de autenticación se encuentra la autenticación biométrica que incluye la detección de una señal biométrica, la extracción de diversas características contenidas en la señal biométrica, y el uso de clasificadores para manejar las características extraídas [14]. La autenticación biométrica es aplicable a distintas características genéticas de los individuos, tales como: iris de los ojos [15], huellas dactilares [16] y el rostro [1].

Los diferentes tipos de software cuentan con una autenticación débil que puede ser vulnerada si se consiguen las credenciales necesarias, es decir que cualquier individuo puede acceder al sistema si cuenta con el usuario y contraseña correctos para ser admitido, debido a que la autenticación se realiza bajo un solo factor[7].Por otra parte se pudo identificar información relevante que debe ser tomada en cuenta para el desarrollo de un sistema de autenticación biométrico facial, elementos como el nivel de iluminación, nivel de brillo del ambiente y perspectiva de la imagen afectan a la recopilación de características de las imágenes, sobre todo si se trabaja bajo un concepto de reconocimiento facial automatizado 2D[1].

En este artículo se tratará de abordar un sistema que permita el reconocimiento facial de una persona para el control de accesos mediante Inteligencia Artificial captados por una videocámara, cubriendo las brechas o limitaciones como son el nivel de iluminación, nivel de brillo del ambiente [1], mediante un trabajo en 3D, además que vamos a realizar múltiples capturas de imagen para incrementar el reconocimiento facial [10] y reducir la repercusión del ambiente.

Materiales y métodos o Metodología computacional

Estado del arte:

A. Un sistema de reconocimiento facial integrado de bajo costo para el control de acceso a puertas mediante Deep Learning

El presente artículo [2] presenta cómo se utiliza un procesador Neural Compute Stick 2 junto con una placa Raspberry Pi 3 B+ para controlar una cerradura electromagnética usando técnicas de aprendizaje profundo para generar un sistema de control de acceso con capacidades de reconocimiento facial integradas desarrollado en el lenguaje de programación Python.

B. Automatizado basado en redes neuronales convolucionales Sistema de Asistencia mediante Reconocimiento Facial Dominio

El presente artículo [7] tiene como finalidad lograr el reconocimiento de rostros en imágenes de video, videos pregrabados o a través de una cámara en vivo para la implementación de un sistema de asistencia automático desarrollado con lenguaje de programación Python que marcará la asistencia solo a los alumnos que hayan asistido a más del 60% de las clases, dejando al resto con inasistencia. Los métodos que implementados son: Análisis de Componentes, análisis de discriminante lineal, patrón binario local y el clasificador de gabo; los cuales reducen la cantidad de cálculos, reducen la complejidad y aumentan la precisión en el reconocimiento facial.

Fundamentación Teórica

Reconocimiento Facial

Identificación de rostros con bajos niveles de sesgo mediante la aplicación de la Inteligencia artificial.[9]

El reconocimiento biométrico facial es definido como una tecnología de inteligencia artificial que implementa comparaciones automáticas de diversos rasgos faciales.[10]

Inteligencia Artificial

La inteligencia artificial (IA) es un conjunto de algoritmos (reglas que definen con precisión un conjunto de operaciones) que permiten realizar cálculos para percibir, razonar y actuar. La IA es usada para llevar a cabo la realización de múltiples tareas, pero puede usarse para brindar mejoras a la inteligencia humana.[8]

Machine learning

Machine learning es definido como aprendizaje automático, y su uso está enfocado en el análisis masivo de datos [19]. Dentro de sus algoritmos más usados tenemos: SVM, BOSQUE ALEATORIO, Árbol de decisiones KNN y Adaboost clasificadores [20].

Deep Learning

Definido como aprendizaje profundo, ofrece una estrategia de optimización global. Dentro de sus usos tenemos: Procesamiento de información, reducción de ruido en imágenes [21], procesamiento del lenguaje natural [22], máquina de traducción [24] e ingeniería de software [25]. Además, el aprendizaje profundo es una rama derivada del aprendizaje automático (Machine learning) [23].

Redes Neuronales

Es un algoritmo inspirado en el cerebro humano diseñado para reconocer patrones en conjuntos de datos numéricos. Los datos del mundo real, por ejemplo, imagen, audio de texto, video, etc. necesita ser transformado en vectores numéricos para usar redes neuronales. Una red neuronal se compone de diferentes capas y una capa se compone de múltiples nodos.

Según el tipo de patrón que la red neuronal está tratando de aprender, a cada dato de entrada que ingresa a un nodo se le asigna cierto peso. Estos pesos determinan la importancia de los datos de entrada para producir el resultado final. Se calcula la suma ponderada de los datos de entrada y, dependiendo de algunos sesgos de umbral, se determina la salida para el nodo. La asignación de entrada a salida se realiza mediante alguna función de activación [17].

Redes Neuronales convolucionales

Son un tipo de red neuronal que se utiliza principalmente en el campo de clasificación de imágenes, particularmente en el reconocimiento facial. Las redes neuronales convolucionales toman una imagen de entrada y modifican los pesos de la red en función de la imagen de entrada para que pueda diferenciarla de otras imágenes. Esto permite que la red aprenda e identifique las características importantes por sí misma. Por

la tanto, se minimiza la necesidad de supervisión humana, reducen la necesidad de procesamiento requerido para entrenar el modelo [7].

Seguridad de la información

Es la disciplina que, con base en políticas y normas internas y externas de la empresa, se encarga de proteger la integridad y privacidad de la información almacenada en un sistema informático, contra cualquier tipo de amenaza, minimizando los riesgos tanto físicos como físicos. lógica, a la que está expuesto.[18]

Control de Accesos

Es implementado como un método de seguridad para delimitar un conjunto de usuarios autorizados para acceder a una determinada información [2].

Herramientas y elementos

Visual Studio Code Visual Studio Code es definido como una plataforma de código abierto, un editor de código de multiplataforma que pertenece a Microsoft y proporciona todos los componentes necesarios de un IDE como: IntelliSense, depuración, control de versiones, creación de plantillas y API de extensiones que brindan muchas facilidades a los desarrolladores.[11]

Python Python es un lenguaje de programación ampliamente utilizado en las aplicaciones web, el desarrollo de software, la ciencia de datos y el machine learning (ML). Los desarrolladores utilizan Python porque es eficiente y fácil de aprender, además de que se puede ejecutar en muchas plataformas diferentes.[6]

Las librerías utilizadas en este lenguaje de programación para este proyecto fueron las siguientes:

Numpy NumPy es un módulo de Python. El nombre es un acrónimo de Python Numérico. Es una librería que consiste en objetos de matrices multidimensionales y una colección de rutinas para procesar esas matrices. Es un módulo de extensión para Python, escrito en su mayor parte en C. Esto asegura que las funciones y funcionalidades matemáticas y numéricas recompiladas de NumPy garantizan una gran velocidad de ejecución.[3]

Os Este módulo provee una manera versátil de usar funcionalidades dependientes del sistema operativo, algunas como leer, escribir, manipular archivos y demás.[4]

OpenCV Es una biblioteca de código abierto que incluye varios cientos de algoritmos de visión artificial.[5]

Imutils Es un paquete de OpenCV que evita la pérdida de fotogramas en el procesamiento de imágenes mediante el uso de múltiples subprocesos que llevan a cabo la lectura y procesamiento de las imágenes en forma simultánea.[12]

Uso del Sistema de detección

Creación de los Datasets

Se usó un dataset creado por un script escrito en python que podemos apreciar en la Figura 1, el cual se encarga de primero trabajar con el frame que es capturado por la cámara para luego evaluarlo por "haarcascade" y detectar la presencia de un rostro, de existir la presencia de algún rostro, es extraído, redimensionado a 720x720 y almacenado como un archivo de formato JPG en una carpeta con el nombre de la persona que estamos registrando, el cual definimos en la línea 3 como *personCode*, se mantendrá en un bucle hasta alcanzar la cantidad de 450 imágenes registradas para posteriormente finalizar el proceso.

Entrenamiento

Revista Innovación y Software
Vol. 4, No. 1, Mes Marzo-Agosto, 2023
ISSN: 2708-0935
Pág. 24-36
https://revistas.ulasalle.edu.pe/innosoft

Se ejecuta un script en python el cual se encarga de usar el algoritmo que será usado para el reconocimiento facial es el conocido como “Local Binary Pattern Histogram”, el cual nos ayuda con el reconocimiento de una persona Figura 2.

```python
import cv2, os, imutils

personCode = 'Jean'
personPath = './Data/' + personCode

if not os.path.exists(personPath):
    os.makedirs(personPath)
    print('Directory created: ', personPath)

cap = cv2.VideoCapture(0, cv2.CAP_DSHOW)
faceClassif = cv2.CascadeClassifier(
    cv2.data.haarcascades + 'haarcascade_frontalface_default.xml')
count = 0

while True:
    ret, frame = cap.read()
    if ret == False: break

    frame = imutils.resize(frame, width = 720)
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    auxFrame = frame.copy()
    faces = faceClassif.detectMultiScale(gray, 1.3, 5)

    for (x,y,h,w) in faces:
        cv2.rectangle(frame, (x,y),(x+w, y+h), (0,255,0), 2)
        rostro = auxFrame[y:y+h, x:x+w]
        rostro = cv2.resize(rostro, (720, 720),
            interpolation=cv2.INTER_CUBIC)
        cv2.imwrite(
            personPath + '/'+personCode+'_{}.jpg'.format(count),
            rostro)
        count+=1

    cv2.imshow('Camera', frame)
    if (cv2.waitKey(1) == 27) or count==450: break

cap.release()
cv2.destroyAllWindows()
```

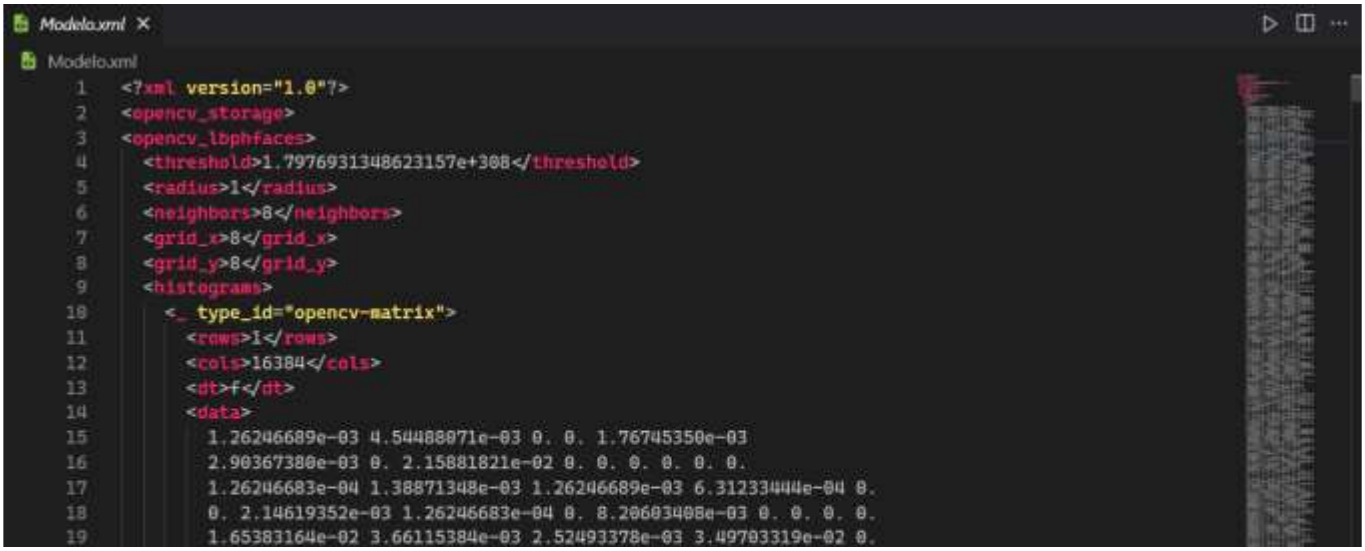
Figura 1. Script para la creación de un dataset personalizado por persona

Lo que haremos es brindarle la data (imágenes generadas anteriormente) junto a sus respectivos labels que no son más que valores asignados a cada persona registrada, que van desde el 0 hasta donde sea necesario, todo esto con el fin de que el algoritmo pueda ser entrenado y finalmente nos genere un “Modelo” en formato xml que contenga todos los valores necesarios para que sea capaz de reconocer a las personas que protagonizan las imágenes entregadas anteriormente.

Revista Innovación y Software
Vol. 4, No. 1, Mes Marzo-Agosto, 2023
ISSN: 2708-0935
Pág. 24-36
https://revistas.ulasalle.edu.pe/innosoft

```python
import cv2, numpy as np, os

dataPath = './Data/'

peopleList = os.listdir(dataPath)

print("Lista de usuarios registrados", peopleList)

labels = []
facesData = []
label = 0

for code in peopleList:
    personPath = dataPath+code
    print("Leyendo Imagenes de :", code)

    for fileName in os.listdir(personPath):
        labels.append(label)

        facesData.append(
            cv2.imread(personPath+'/'+fileName, 0))

    label+=1

cv2.destroyAllWindows()

faceRecognizer = cv2.face.LBPHFaceRecognizer_create()
#Entrenamiento
print("Entrenando")
faceRecognizer.train(facesData, np.array(labels))
print("Modelo Entrenado")
#GuardarModelo
faceRecognizer.write('Modelo.xml')
print("Modelo guardado")
```

Figura 2. Script para el entrenamiento de la red con los datos de los individuos registrados



```
Modelo.xml
<?xml version="1.0"?>
<opencv_storage>
<opencv_lbphfaces>
  <threshold>1.7976931348623157e+308</threshold>
  <radius>1</radius>
  <neighbors>8</neighbors>
  <grid_x>8</grid_x>
  <grid_y>8</grid_y>
  <histograms>
    <_ type_id="opencv-matrix">
      <rows>1</rows>
      <cols>16384</cols>
      <dt>f</dt>
      <data>
        1.26246689e-03 4.54488071e-03 0. 0. 1.76745350e-03
        2.90367380e-03 0. 2.15881821e-02 0. 0. 0. 0. 0. 0.
        1.26246683e-04 1.38871348e-03 1.26246689e-03 6.31233444e-04 0.
        0. 2.14619352e-03 1.26246683e-04 0. 8.20603408e-03 0. 0. 0. 0.
        1.65383164e-02 3.66115384e-03 2.52493378e-03 3.49703319e-02 0.
```

Figura 3. Modelo que contiene los valores numéricos para que la red sea capaz de reconocer a los individuos registrados

El modelo generado contiene los valores de los pesos y sesgos necesarios en la red para que ésta sea capaz de reconocer a las personas que han sido registradas en el entrenamiento, no es necesario que comprendamos

Revista Innovación y Software
Vol. 4, No. 1, Mes Marzo-Agosto, 2023
ISSN: 2708-0935
Pág. 24-36
https://revistas.ulasalle.edu.pe/innosoft

cada valor que está incluido en este modelo, simplemente con comprender que en su conjunto está personalizado para el reconocimiento facial de aquellos individuos que registramos con su respectivo dataset. Figura 3.

Predicción

Una vez entrenado, el modelo está listo para realizar predicciones, para ello desarrollamos un script de python el cual se encarga de primero trabajar con el frame que es capturado por la cámara para luego evaluarlo por "haarcascade" y detectar la presencia de un rostro, de existir la presencia de algún rostro, es extraído, redimensionado a 720x720, y este último será el frame que ingresará al modelo para poder predecir y que este nos devuelva su predicción. Figura 4.

Figura 4. Script para la predicción usando el modelo entrenado

Revista Innovación y Software
Vol. 4, No. 1, Mes Marzo-Agosto, 2023
ISSN: 2708-0935
Pág. 24-36
https://revistas.ulasalle.edu.pe/innosoft

La predicción del modelo nos devuelve 2 valores, primero el número del label con el cual tiene mejor coincidencia y en segundo lugar, la distancia o que tan alejado se encuentra el frame actual de los registrado para el entrenamiento que pertenecen a dicho label, el nivel de confiabilidad puede ser personalizado dependiendo del nivel de confianza que se desea en el momento de la autenticación, en la línea 37 se estipula la lejanía máxima a la que se puede encontrar la imagen de entrada de la predicción para ser aceptado como reconocido, mientras que en la línea 17, el valor de maxTime indica la cantidad de frames máximos que se evaluarán para la predicción, por su parte en la línea 32 se establece que el mínimo de frames en los que debe ser reconocido el individuo para considerarlo como tal, reconocido, 0.8(80%), luego de que se cumpla el maxTime, se pinta en pantalla el resultado, sea positivo o negativo durante unos segundos para luego finalizar su ejecución.

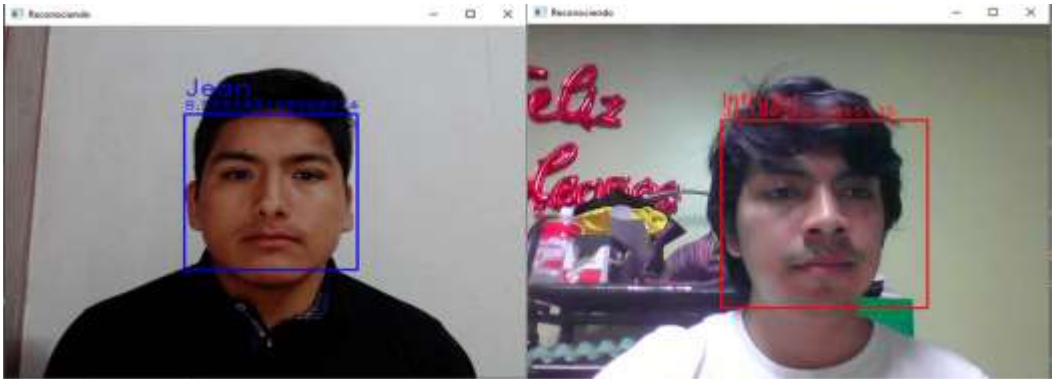
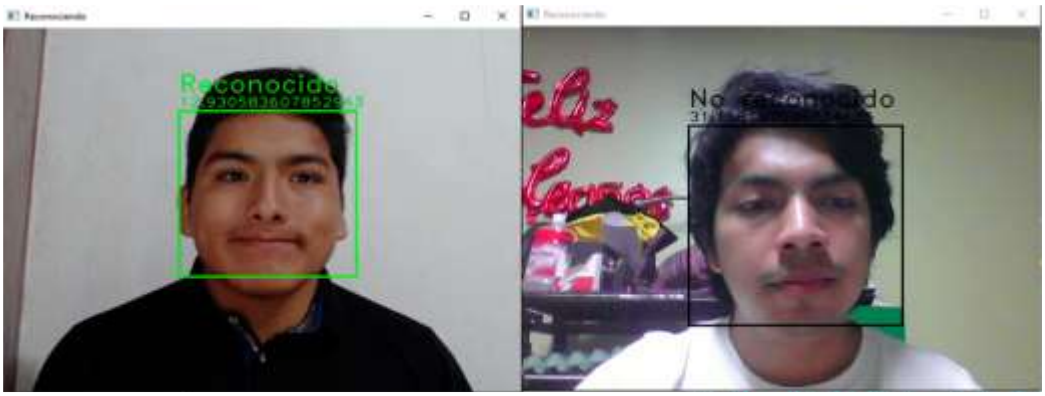


Figura 5. Ejecución del script de reconocimiento

En la Figura 5 podemos visualizar la apariencia de la ejecución del script de reconocimiento tanto cuando reconoce cuando al individuo, como cuando no.



Revista Innovación y Software
Vol. 4, No. 1, Mes Marzo-Agosto, 2023
ISSN: 2708-0935
Pág. 24-36
https://revistas.ulasalle.edu.pe/innosoft

Figura 6. Proyección de resultados positivos y negativos de autenticación

En la Figura 6 podemos observar cómo se muestran los resultados finales de autenticación, tanto positiva como negativa.

Resultados y discusión

Los parámetros que usa el sistema pueden ser perfectamente ajustados de acuerdo a lo que uno necesite, en este caso se usaron 450 imágenes por persona registrada, pero a mayor este número, mayor será la precisión del sistema.

Respecto a la precisión, al realizar pruebas se obtiene como segundo valor devuelto, que tan alejado está el frame de la predicción hecha por el sistema, al sacar una media de estos valores devueltos, se obtuvo como resultado el valor de 12.070609534085545, esto se traduce en un porcentaje de acierto de un 88% aproximadamente en cuanto a las predicciones por persona se refiere, Figura 8.

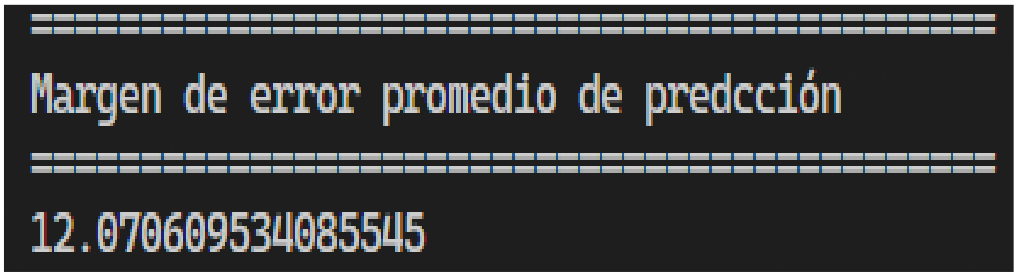


Figura 8. Margen de error promedio de predicción

Mientras que el porcentaje de aciertos totales dentro de una prueba unitaria fue considerado como un valor mínimo a alcanzar para validar la autenticación de la persona, en este caso como se indicó previamente, es del 80%, al igual el tamaño unitario de los datasets, se puede ajustar manualmente.

Conclusiones

El sistema de reconocimiento se puede ajustar para una mayor eficiencia siempre y cuando se incremente el tamaño de los datasets generados por individuo para un posterior más robusto entrenamiento.

El lenguaje de programación Python brinda muchas facilidades para la programación de este tipo, refiriéndonos a Visión Artificial como a Inteligencia artificial, con la gran cantidad de librerías que ofrecen métodos para una mejor experiencia en estos campos. Algunos rasgos faciales pueden ser compartidos por una o más personas, para ello es mejor ampliar el tamaño de los datasets y así poder ser más específicos al entrenar a la red neuronal. Los datasets de imágenes solo son usados para crear el modelo, una vez esto esté hecho ya no es necesario mantenerlos, puede deshacerse de ellos o almacenarlos, de aquí en adelante para las predicciones sólo es necesario usar los valores que están dentro del modelo generado.

Referencias

[1] "Facial Expression Recognition Using Machine Learning Techniques", *International Journal of Advance Engineering and Research Development*, vol. 1, n.º 06, junio de 2020. Accedido el 17 de noviembre de 2022. [En línea]. Disponible: https://doi.org/10.21090/ijaerd.010633

[2] R. F. Rahmat, E. N. Zai, I. Fawwaz y I. Aulia, "Facial Recognition-Based Automatic Door Access System Using Extreme Learning Machine", *IOP Conference Series: Materials Science and Engineering*, vol. 851, p. 012065, mayo de 2020. Accedido el 17 de noviembre de 2022. [En línea]. Disponible: https://doi.org/10.1088/1757-899x/851/1/012065

[3] "Librería NumPy - 🤖 Aprende IA". 🤖 Aprende IA. https://aprendeia.com/libreria-de-python-numpy-machine-learning/ (accedido el 17 de noviembre de 2022).

[4] "os - Interfaces miscelíneas del sistema operativo — documentación de Python - 3.10.8". 3.11.0 Documentation. https://docs.python.org/es/3.10/library/os.html (accedido el 17 de noviembre de 2022).

[5] "OpenCV: OpenCV modules". OpenCV documentation index. https://docs.opencv.org/4.x/ (accedido el 17 de noviembre de 2022).

[6] "¿Qué es Python? | Guía de Python para principiantes de la nube | AWS". Amazon Web Services, Inc. https://aws.amazon.com/es/what-is/python/ (accedido el 17 de noviembre de 2022).

[7] Anuja Jadhav, Yash Joshi y Vishakha Kalambe, "Face Based Attendance System Using Convolutional Neural Network", *International Journal of Advanced Research in Science, Communication and Technology*, pp. 51–54, febrero de 2022. Accedido el 17 de noviembre de 2022. [En línea]. Disponible: https://doi.org/10.48175/ijarsct-2506

[8] O. Niel y P. Bastard, "Artificial Intelligence in Nephrology: Core Concepts, Clinical Applications, and Perspectives", *American Journal of Kidney Diseases*, vol. 74, n.º 6, pp. 803–810, diciembre de 2019. Accedido el 17 de noviembre de 2022. [En línea]. Disponible: https://doi.org/10.1053/j.ajkd.2019.05.020

[9] T. Walsh, "The troubling future for facial recognition software", *Communications of the ACM*, vol. 65, n.º 3, pp. 35–36, marzo de 2022. Accedido el 15 de noviembre de 2022. [En línea]. Disponible: https://doi.org/10.1145/3474096

[10] M. Smith y S. Miller, "The ethical application of biometric facial recognition technology", *AI & SOCIETY*, abril de 2021. Accedido el 17 de noviembre de 2022. [En línea]. Disponible: https://doi.org/10.1007/s00146-021-01199-9

[11] S. Latifi, Ed., *17th International Conference on Information Technology–New Generations (ITNG 2020)*. Cham: Springer International Publishing, 2020. Accedido el 17 de noviembre de 2022. [En línea]. Disponible: https://doi.org/10.1007/978-3-030-43020-7

[12] S. K. Shammi, S. Sultana, M. S. Islam y A. Chakrabarty, "Low Latency Image Processing of Transportation System Using Parallel Processing co-incident Multithreading (PPcM)", en *2018 Joint 7th International Conference on Informatics, Electronics & Vision (ICIEV) and 2018 2nd International Conference on Imaging, Vision & Pattern Recognition (icIVPR)*, Kitakyushu, Japan, 25–29 de junio de 2018. IEEE, 2018. Accedido el 17 de noviembre de 2022. [En línea]. Disponible: https://doi.org/10.1109/iciev.2018.8640957

[13] M. Nicho, S. Khan y M. S. M. K. Rahman, "Managing Information Security Risk Using Integrated Governance Risk and Compliance", en 2017 International Conference on Computer and Applications (ICCA), Doha, United Arab Emirates, 6–7 de septiembre de 2017. IEEE, 2017. Accedido el 17 de noviembre de 2022. [En línea]. Disponible: https://doi.org/10.1109/comapp.2017.8079741

[14] S. Rasnayaka, S. Saha y T. Sim, "Making the most of what you have! Profiling biometric authentication on mobile devices", en *2019 International Conference on Biometrics (ICB)*, Crete, Greece, 4–7 de junio de 2019. IEEE, 2019. Accedido el 17 de noviembre de 2022. [En línea]. Disponible: https://doi.org/10.1109/icb45273.2019.8987402

[15] I. Sluganovic, M. Roeschlin, K. B. Rasmussen y I. Martinovic, "Analysis of Reflexive Eye Movements for Fast Replay-Resistant Biometric Authentication", *ACM Transactions on Privacy and Security*, vol. 22, n.º 1, pp. 1–30, enero de 2019. Accedido el 17 de noviembre de 2022. [En línea]. Disponible: https://doi.org/10.1145/3281745

[16] L. Monastyrskii, V. Lozynskii, Y. Boyko y B. Sokolovskii, "Fingerprint recognition in inexpensive biometric system", *Electronics and Information Technologies*, vol. 9, 2018. Accedido el 17 de noviembre de 2022. [En línea]. Disponible: https://doi.org/10.30970/eli.9.120

Roles de Autoría

Jean Elias Manuel Reyes Campos: Conceptualización, Curación de datos, Investigación, Metodología, Software, Validación, Redacción - borrador original. **Christian Stephano Castañeda Rodríguez:** Conceptualización, Curación de datos, Investigación, Metodología, Software, Validación, Redacción - borrador original. **Luis Daniel Alva Luján:** Conceptualización, Curación de datos, Investigación, Metodología, Software, Validación, Redacción - borrador original. **Alberto Carlos Mendoza de los Santos:** Investigación, Metodología, Software, Supervisión, Validación, Redacción - borrador original.