# Bloqueo de puerta con reconocimiento de rostro y Arduino

Asunción-Pomasonco, Alexia Nicoll[1] Inca-Huamaní, Brian Omar[2]

Universidad Nacional Tecnológica de Limar Sur[1,2]
2113110163@untels.edu.pe [1]
2113110074@untels.edu.pe [2]

**Abstract**

The citizen insecurity panorama in Peru reveals persistent challenges, notably in residential and business thefts, including Micro and Small Enterprises (MYPES). Addressing the growing need for asset protection, we propose a facial recognition-based door lock prototype integrating computer vision and Python-Arduino programming. Using the Haar Cascade algorithm for facial detection and LBPH method for recognition, the system achieved a 92% accuracy rate. This research not only addresses current security concerns but also sets the stage for potential advancements. Future work recommendations include real-world implementation and exploration of advanced techniques such as convolutional networks, machine learning, and alternative algorithms represents a promising avenue for further enhancing the system's capabilities and resilience against evolving security challenges.
**Keywords:** reconocimiento, rostro, arduino, algoritmo.

## Introducción

En el contexto actual del Perú, el robo a viviendas y hogares continúa siendo uno de los principales problemas relacionados con la inseguridad ciudadana. De acuerdo con información del Instituto Nacional de Estadística e Informática (INEI), el porcentaje promedio de viviendas afectadas por robo en los últimos 3 años es del 4.5%, habiéndose incrementado un 0.9% a comparación del año 2022. Así mismo, durante los años 2015 y 2016, el intento de robo y robo a viviendas fue de un 12% y, aunque este porcentaje ha disminuido gradualmente hasta alcanzar un 8.8% en 2021, se observó un repunte en el primer semestre de 2022 con una cifra del 10.1% (Instituto Nacional de Estadística e Informática, 2021).
Esta situación no se limita solo a los hogares, ya que los negocios, incluyendo las Micros y Pequeñas Empresas (MYPES), también se ven afectados. Durante diciembre de 2022, el 74.65% de los robos ocurrieron en negocios, mientras que el 25.35% restante tuvo lugar en viviendas. Este panorama plantea la necesidad apremiante de salvaguardar tanto los bienes de valor en los hogares como los implementos de trabajo y suministros en los negocios (Gestión, 2023).
En respuesta a esta creciente inseguridad en espacios privados, surge la propuesta de este trabajo: un prototipo de bloqueo de puertas basado en el reconocimiento facial. Esta iniciativa busca integrar la visión computacional con la programación en Python y Arduino para desarrollar un sistema de seguridad eficiente, con el propósito de mitigar el acceso no autorizado y fortalecer la protección de los bienes en viviendas y negocios.
A continuación, se describe los principales antecedentes usados en la investigación para justificar el presente artículo.

Table 1: *Antecedentes*

| **Proposal** | **Techniques** | **Results** | **Ref.** |
|---|---|---|---|
| Propuesta de un algoritmo de Seguimiento de actitud eficiente para el reconocimiento facial | OpenCV | 96.23% de eficiencia. | (Zhu & Cheng, 2020) |
| Desarrollo de un sistema de reconocimiento de personas basado en el reconocimiento facial | Haar Cascade | Identificación exitosa con un leve margen de error. | (León et al., 2022) |

Continues on the next page

Table 1 – Continued

| Proposal | Techniques | Results | Ref. |
|---|---|---|---|
| Desarrollo de un sistema de control de acceso automático basado en reconocimiento facial | HOG, SVM, FaceNet | 97% de acierto | (Rameswari, Kumar, Aananth, & Deepak, 2021) |
| Desarrollo de un sistema de reconocimiento facial para permitir el registro y control de acceso a los empleados y visitantes | Redes neuronales artificiales y métodos basados en características geométricas | Permite la monitorización de las fechas y horas de ingreso a través de la generación de reportes. | (Ibarra-Estévez & Paredes, 2018) |
| Desarrollo de un sistema de seguridad para el bloqueo y desbloqueo de una puerta magnética basado en reconocimiento facial. | OpenCV, algoritmos Eigenfaces y PCA | Alcanzó el 90% de acierto | (Gunawan, Hasan Gani, Abdul Rahman, & Kartiwi, 2017) |
| Implementación de un sistema de bloqueo y desbloqueo de una puerta automática a través de detección facial. | Algoritmos Haar Cascade y LPBH | El sistema logró reconocer el rostro en un 97.2% a una distancia óptima de 1.5 metros. | (A et al., 2021) |
| Desarrollo de un sistema de cerradura de puerta basada en el reconocimiento facial | LBPH | Cuando se detecta el rostro, se autoriza el acceso; de lo contrario, se solicita ingresar un PIN manual para desbloquear la puerta. | (Vamsi, Sai, & Vijayalakshmi, 2019) |
| Desarrollo de un nuevo método de identificación facial para implementarse en sistemas de cerraduras biométricas. | Redes neuronales convolucionales (Facenet) | El sistema de identificación facial logra una tasa de reconocimiento correcto superior al 99% y un alto rendimiento. | (Burruel, Rangel, Peralta, González, & Morales, 2021) |
| Implementación de un prototipo de puerta automática basada en reconocimiento facial. | Extreme Learning Machine | Alcanzó un 70% de exactitud en la detección de los rostros con 10 capas ocultas, 85% con 30 capas ocultas y 90% con 50 capas ocultas. | (Rahmat, Zai, Fawwaz, & Aulia, 2020) |
| Implementación de cerradura de puerta solenoide con reconocimiento facial | OpenCV, algoritmos Haar Cascade y LBPH | Los resultados fueron satisfactorios. | (Deep, Singh, & Singh, 2022) |

# Justificación

El presente proyecto se lleva a cabo para implementar un sistema innovador de bloqueo de puertas basado en reconocimiento facial, aprovechando los éxitos demostrados por estudios previos. La alta precisión alcanzada en tasas de reconocimiento facial superiores al 99% respalda la viabilidad de esta tecnología. Además, la identificación de desafíos ambientales, como la iluminación y el tamaño, impulsa el desarrollo de estrategias para mitigar estos efectos. La optimización de hardware, en particular, el uso de Arduino, se plantea para garantizar un rendimiento eficiente y ágil. La diversidad de algoritmos previamente empleados proporciona una oportunidad para seleccionar el más adecuado, equilibrando velocidad y precisión. La seguridad y usabilidad se abordan mediante medidas robustas y una interfaz de usuario intuitiva. La integración de tecnologías adicionales y la optimización del tiempo de procesamiento completan la justificación, estableciendo este proyecto como un avance significativo en la implementación práctica de sistemas de bloqueo de puertas basados en reconocimiento facial.

# Materiales y Métodos

## Materiales

- Cámara web: Cámara Web Enkore Geneve – ENC WC 112. Resolución: 640 x 480 px. Formato de Vídeo: 24 Bits RGB a 30 fps.
- Arduino: Simulación del Arduino UNO en Proteus.
- Lenguaje de programación Python: Uso de Python con las bibliotecas opencv-python y serial instaladas.

## Métrica de precisión

Para medir la precisión del sistema de acceso, se plantea la siguiente métrica que consiste en la fracción de intentos satisfactorios de detección facial sobre el total de intentos que se hayan realizado.

$$\text{Precisión} = \frac{\text{Intentos satisfactorios}}{\text{Intentos totales}} \times 100\%$$

## Entrenamiento del modelo

Para el entrenamiento del modelo se optó por el método LBPH, el cual es el resultado de la fusión del método LBP (Local Binary Pattern), el cual asigna etiquetas a los píxeles de una imagen al evaluar el umbral de la vecindad de cada píxel y trata el resultado como un valor binario, con histogramas, consiguiendo así la expresión de las imágenes faciales mediante un vector de datos simple (Verdeguer-Valderrama & Campos-Vasquez, 2021). Para ello, se necesitó vídeos donde se muestren los rostros de cada persona desde diferentes ángulos e iluminación para hacerlo consistente, donde posteriormente a través de un bucle iterativo se toman los frames, se almacenan en un arreglo Numpy y se procede a crear el modelo *'modeloLBPHFace.xml'* que se utilizará en el proyecto.

## Metodología

La metodología propuesta se describe en la figura 1, la cual representa el flujo de información del rostro detectado a través de la cámara web utilizando el lenguaje de programación Python y Arduino para el reconocimiento facial y control de acceso en el sistema de seguridad.
El proceso inicia al escanear el rostro de una persona por medio de la cámara web, donde posteriormente se realiza la detección facial mediante el uso del algoritmo de Haar Cascade, luego se pasa a reconocer el rostro captado con los rostros permitidos en una base de datos con el método LBPH (Local Binary Pattern Histogram) y finalmente se realiza la decisión de permitir o no el ingreso al ambiente por el sistema.

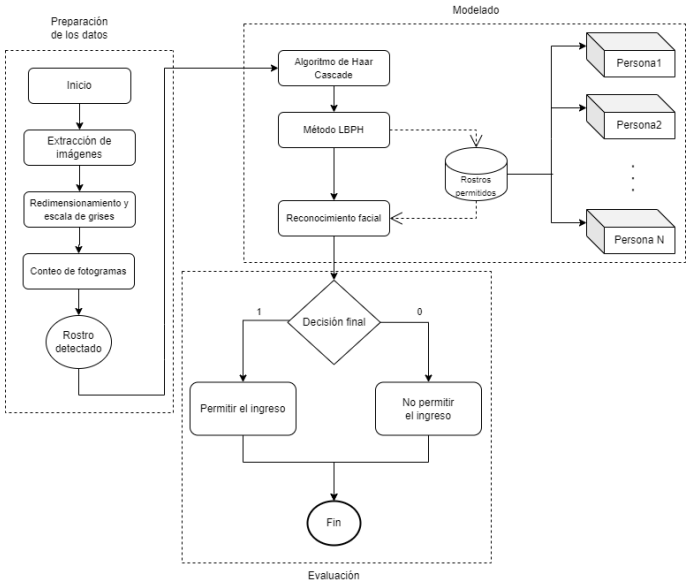


Figure 1: Diagrama de la investigación

- Preparación de los datos

    1. Extracción de imágenes: Se capturan las imágenes del rostro mediante la cámara web. Para ello, se utiliza la siguiente función en OpenCV:
    *cv2.VideoCapture(1, cv2.CAP_DSHOW)*
    Donde el argumento (1) puede variar en función al número de dispositivos de cámara que puedan estar conectados al ordenador y el argumento cv2.CAP_DSHOW indica que utiliza la interfaz DirectShow para la captura de video.
    2. Redimensionamiento y escala de grises: La conversión a escala de grises es el proceso de convertir imágenes RGB en escala de grises. Dicho procedimiento se realiza con la función de OpenCV:
    *cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)*
    Donde el argumento COLOR_BGR2GRAY indica la conversión de la imagen RGB a grises.
    El redimensionamiento implica la obtención de las imágenes de un mismo tamaño. Para el estudio, el tamaño específico de las imágenes es de 100 x 100 píxeles.
    En OpenCV, el redimensionamiento se aplica utilizando la siguiente función:
    *face_cascade.detectMultiScale(gray, scaleFactor=1.2, minNeighbors=4, minSize=(100, 100))*
    Los argumentos son los siguientes:
        * *gray* detecta imágenes en escala de grises
        * *scaleFactor* especifica cuánto se reduce el tamaño de la imagen en cada escala de imagen
        * *minNeighbors* especifica cuántos vecinos debe tener cada rectángulo candidato para retenerlo.
        * *minSize* se usa para fijar el tamaño mínimo posible para la detección.
    3. Conteo de fotogramas: Se procede a leer los fotogramas detectados por la cámara web con la función:
    *cap.read()*

- Modelado: Detección de rostro

    1. Algoritmo de Haar Cascade: Propuesto por Viola y Jones, consiste en una técnica de clasificación en cascada que combina varios clasificadores débiles, donde cada clasificador examina una sección específica de una imagen o fotograma en un video. El proceso implica la organización de la imagen mediante la evaluación de características vinculadas a rectángulos o bloques de píxeles en ventanas de detección cerradas. Estas ventanas tienen tamaños idénticos y contienen 2, 3 y 4 rectángulos, aplicándose la función Haar en cada una. La función Haar calcula la diferencia entre la suma de los píxeles en rectángulos blancos y la suma en un rectángulo sombreado. Este enfoque permite identificar las características del objeto al comparar resultados entre imágenes positivas y negativas, buscando similitudes en todas las imágenes positivas (Viola & Jones, 2001).
    Para la investigación, se procede a cargar el archivo pre-entrenado de detección de rostros *haarcascade_frontalface_default.xml* del algoritmo Haar Cascade. Este archivo nos permitirá leer y detectar rostros de manera eficaz. Para cargar el archivo se utiliza la función de OpenCV:
    *cv2.CascadeClassifier(cv2.data.haarcascades+'haarcascade_frontalface_default.xml')*
    2. Método LBPH: Inicializamos el uso del método LBPH a través de la sentencia:
    *cv2.face.LBPHFaceRecognizer_create()*
    Luego procedemos a cargar el modelo entrenado mediante la siguiente función:
    *face_recognizer.read('../face-py-ino-proteus/model/modeloLBPHFace.xml')*
    Para su ejecución en el programa, primero se deberá determinar las dimensiones de la imagen del rostro a captar mediante la siguiente función en OpenCV: *cv2.resize(rostro, (150, 150), interpolation=cv2.INTER_CUBIC)*, donde se indica que tendrá dimensiones de 150x150 píxeles.
    Posteriormente, se declara la función para realizar el reconocimiento facial:
    *face_recognizer.predict(rostro)*
    3. Reconocimiento facial: Para alcanzar el propósito se realizó una comparativa previa del rostro captado con los rostros almacenados en el modelo entrenado con el método LBPH y, a su vez, se establece que debe alcanzar los 40 fotogramas de imagen para poder dar un veredicto con respecto al acceso.

- Modelado: Simulación en Proteus Para realizar el prototipo del modelado del Arduino UNO en Proteus, se siguieron los siguientes pasos.

    1. Virtual Serial Port Driver: Este programa nos permitirá crear los puertos virtuales para conectar el Proteus con el lenguaje de programación Python. Específicamente, el puerto COM1 será para el dispositivo COMPIM en Proteus, y el puerto COM2 para el lenguaje Python.
    2. Proteus: Para tal finalidad, se necesitan los siguientes componentes:
        a. Simulino Uno: Es un modelo de simulación en Proteus que representa la placa de desarrollo Arduino Uno.
        b. COMPIM: Es un componente en Proteus que simula una interfaz de comunicación serial.
        c. Pantalla LCD (LM016L): Permite a los diseñadores emular la interfaz y el comportamiento de una pantalla LCD.
        d. Potenciómetro (POT-HG): Es un componente electrónico ajustable que se utiliza para variar la resistencia eléctrica en un circuito.

    La relación entre los componentes en proteus se muestra en la Figura 2.

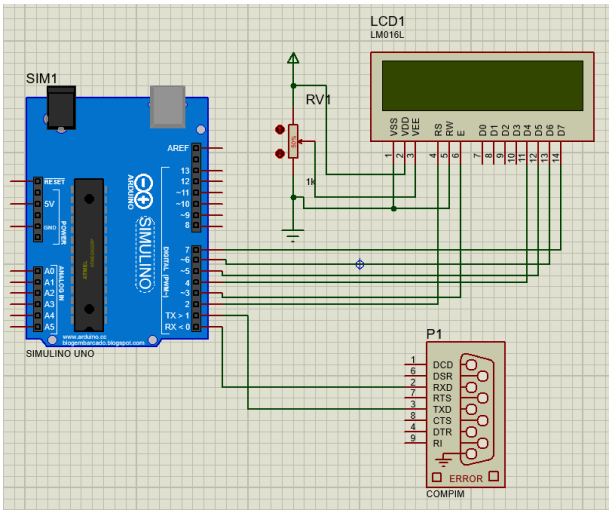


Figure 2: Relación de los componentes en Proteus

## Resultados

La implementación del sistema en base al lenguaje Python y Arduino parte mostrando una ventana que muestra las imágenes captadas por la cámara web del sistema como se muestra en la figura 3.

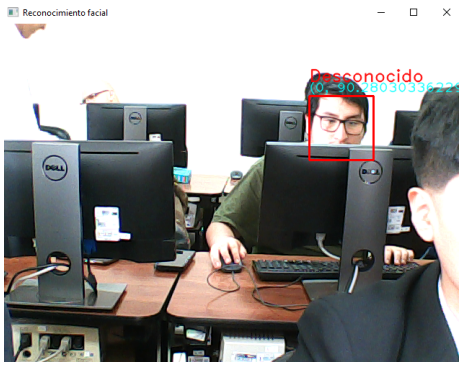


Figure 3: Ventana mostrando las imágenes captadas por la cámara web

Al captar un rostro, se dibujará un cuadrado alrededor de este para realizar su debida identificación, véase en la figura 4.

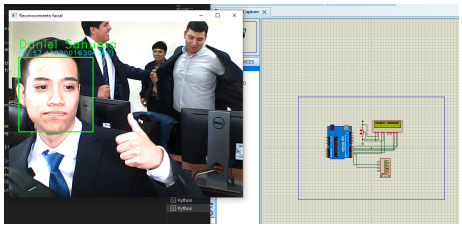

Figure 4: Dibujo de un cuadrado alrededor del rostro

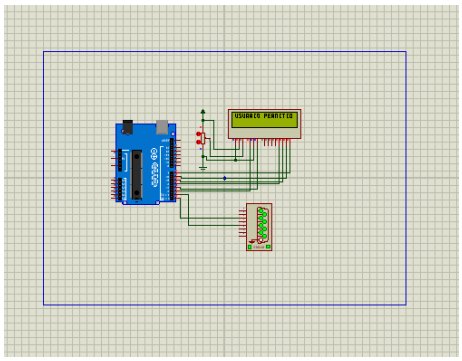


Figure 5: Arduino permitiendo el acceso

Si el rostro captado forma parte de los rostros permitidos, se permitirá el acceso mediante la comunicación con Arduino y se mostrará por una pantalla LCD en el modelo de Proteus, véase en la figura 5.

El proyecto mostró resultados favorables al detectar y reconocer los rostros con un 92% de eficacia, demostrando así que tiene un buen rendimiento en el entorno de simulación en Proteus, por lo que se evidencia su potencial de aplicación en un entorno real.

## Discusiones

La motivación del trabajo fue realizar una simulación de cerradura de puerta con reconocimiento facial mediante el lenguaje Python y Arduino. Según los resultados se llegó a implementar satisfactoriamente el sistema haciendo uso del algoritmo Haar Cascade y el método LBPH, de esta manera concordando con Rahmat et al (2020) en la eficacia de este último método. Así como en el estudio realizado por A. et al (2021), también se corrobora el potencial de ambos algoritmos, ya que se logró implementar un sistema de cerradura de puerta, además del uso de redes convolucionales, las cuales dieron un valor agregado a la detección facial pues se llegó a un 97.2% de acierto.

## Conclusiones

El panorama de inseguridad ciudadana en el Perú refleja la persistencia de problemas significativos, especialmente en lo que respecta al robo a viviendas, hogares y negocios, incluyendo las Micros y Pequeñas Empresas (MYPES). La creciente necesidad de salvaguardar tanto los bienes personales en hogares como los implementos de trabajo en negocios es evidente a través de las estadísticas que revelan la magnitud de los robos en ambos entornos.

En respuesta a este desafío, surgió la propuesta de un prototipo de bloqueo de puertas basado en el reconocimiento facial, integrando tecnologías como la visión computacional y la programación en Python y Arduino. Esta investigación hizo uso del algoritmo Haar Cascade para la detección facial y el método LBPH para el reconocimiento, mostrando su eficacia pues se llegó a un 92% de acierto.

Como recomendaciones a futuros trabajos se propone la realización del sistema en un entorno real, así como la implementación de técnicas que incluyen redes convolucionales, Learning Machine, así como algoritmos alternos a los utilizados.

# References


A, I. H., P., R., Rahayu, W., Febrianty, D., Farihah, N., Azizah, W., ... Gultom, Y. (2021). Face detection and recognition in real-time photos with haar cascade and local binary pattern histogram for automatic door locking system. In A. M. Nasution, R. A. Wahyuono, & A. M. Hatta (Eds.), *Fourth international seminar on photonics, optics, and its applications (isphoa 2020).* SPIE. doi: 10.1117/12.2587028

Burruel, J. M., Rangel, H. R., Peralta, G. E., González, V. A., & Morales, L. A. (2021). Sistema de control de acceso mediante identificación facial usando aprendizaje profundo. *Research in Computing Science*, *150*(6), 215-227.

Deep, A., Singh, B., & Singh, R. (2022). Face recognition door lock system using raspberry pi. *International Journal for Research in Applied Science and Engineering Technology*, *10*(5), 1733–1735. doi: 10.22214/ijraset.2022.42663

Gestión. (2023). *Los robos en hogares y negocios aumentaron un 82% durante el 2022* (Tech. Rep.). Jr. Jorge Salazar Aráoz N° 171, Santa Catalina, La Victoria, Lima: Author.

Gunawan, T. S., Hasan Gani, M. H., Abdul Rahman, F. D., & Kartiwi, M. (2017). Development of face recognition on raspberry pi for security enhancement of smart home system. *Indonesian Journal of Electrical Engineering and Informatics (IJEEI)*, *5*(4). doi: 10.52549/ijeei.v5i4.361

Ibarra-Estévez, J., & Paredes, K. (2018). Redes neuronales artificiales para el control de acceso basado en reconocimiento facial. *Revista PUCE*. doi: 10.26807/revpuce.v0i106.140

Instituto Nacional de Estadística e Informática. (2021). *Victimización en el perú 2015 - 2021. iii. robo en vivienda* (Tech. Rep.). Av. General Garzón N° 654 - 658 - Jesús María: Author.

León, R., Rodriguez, M., Villegas, G., Honorio, L., Ruiz, J., & Contreras, J. (2022). Sistema de reconocimiento facial para control de seguridad en el ingreso a las empresas. *INGnosis UCV*, *8*, 43-51. doi: https://doi.org/10.18050/ingnosis

Rahmat, R. F., Zai, E. N., Fawwaz, I., & Aulia, I. (2020). Facial recognition-based automatic door access system using extreme learning machine. *IOP Conference Series: Materials Science and Engineering*, *851*(1), 012065. doi: 10.1088/1757-899x/851/1/012065

Rameswari, R., Kumar, S. N., Aananth, M. A., & Deepak, C. (2021). Automated access control system using face recognition. *Materials Today: Proceedings*, *45*, 1251-1256. (International Conference on Advances in Materials Research - 2019) doi: https://doi.org/10.1016/j.matpr.2020.04.664

Vamsi, T., Sai, K. C., & Vijayalakshmi, M. (2019). Face recognition based door unlocking system using raspberry pi. *International Journal of Advanced Research, Ideas and Innovation in Technology*, *5*(2).

Verdeguer-Valderrama, D., & Campos-Vasquez, N. (2021). Diseño e implementación de un sistema de identificación de personas para la seguridad de los accesos a condominios, basado en el algoritmo de reconocimiento facial lbph faces. In *Proceedings of the 19th laccei international multi-conference for engineering, education, and technology: "prospective and trends in technology and skills for sustainable social development" "leveraging emerging technologies to construct the future"*. Latin American and Caribbean Consortium of Engineering Institutions. doi: 10.18687/laccei2021.1.1.213

Viola, P., & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 ieee computer society conference on computer vision and pattern recognition. cvpr 2001.* IEEE Comput. Soc. doi: 10.1109/cvpr.2001.990517

Zhu, Z., & Cheng, Y. (2020). Application of attitude tracking algorithm for face recognition based on opencv in the intelligent door lock. *Computer Communications*, *154*, 390-397. doi: https://doi.org/10.1016/j.comcom.2020.02.003