

**UNIVERSIDAD DEL VALLE DE GUATEMALA**

CC3085 – INTELIGENCIA ARTIFICIAL

Sección 10

Ing. Alberto Suriano



**PROYECTO FINAL**

IMPLEMENTACIÓN DE RECONOCIMIENTO FACIAL

Michelle Angel de María Mejía Villela, 22596

Silvia Alejandra Illescas Fernández, 22376

**Guatemala, 26 de mayo del 2025**

## **1. Descripción del Problema:**

La autenticación en los sistemas informáticos es esencial para garantizar la seguridad y la privacidad de los usuarios. Tradicionalmente, se ha utilizado la autenticación por contraseñas, pero este método presenta varias desventajas. Entre los problemas más comunes están el olvido de contraseñas, la vulnerabilidad ante ataques de suplantación de identidad y la posibilidad de ataques por fuerza bruta. Además, los usuarios tienden a utilizar contraseñas débiles o a reutilizarlas en múltiples sitios, lo que aumenta la probabilidad de ser hackeados.

En este contexto, el reconocimiento facial surge como una alternativa más robusta y cómoda, permitiendo el acceso a los sistemas sin la necesidad de recordar contraseñas. Sin embargo, la implementación de un sistema de reconocimiento facial fiable y eficiente requiere considerar factores como la precisión, el rendimiento y la seguridad. Este proyecto tiene como objetivo desarrollar un sistema de autenticación mediante reconocimiento facial, implementando dos enfoques diferentes: uno utilizando la librería `face_recognition` y otro basado en técnicas tradicionales de procesamiento de imágenes como los histogramas.

## **2. Análisis del Problema:**

Se exploran las dos soluciones propuestas:

Login con librería (`face_recognition`):

Ventajas:

- **Simplicidad en la implementación:** Al utilizar una librería ya existente, como `face_recognition`, se reduce significativamente el tiempo de desarrollo. Esta librería ya contiene implementaciones eficientes para la detección y el reconocimiento facial.
- **Alta precisión:** La librería `face_recognition` se basa en modelos preentrenados y avanzados, lo que garantiza una alta precisión en la identificación de rostros, incluso en condiciones de baja calidad de imagen.
- **Rendimiento optimizado:** La librería está diseñada para procesar imágenes rápidamente, lo cual es crucial para ofrecer una experiencia de usuario fluida y eficiente.

Desventajas:

- **Dependencia de librerías externas:** Usar `face_recognition` significa que el sistema depende de una librería externa, lo cual podría presentar problemas de compatibilidad con futuras versiones de Python o cambios en la librería misma.

- Mayor uso de recursos: El uso de modelos avanzados de reconocimiento facial, aunque eficiente, puede ser más demandante en términos de recursos computacionales. Esto podría afectar el rendimiento en dispositivos con limitaciones de hardware.

Login sin librería (basado en histogramas):

Ventajas:

- Independencia de librerías externas: Este enfoque no requiere ninguna librería externa, lo que lo hace adecuado para entornos donde las dependencias externas no son viables.
- Mayor control: El sistema es completamente personalizable. Al implementar el propio algoritmo de comparación, el sistema permite modificar y ajustar cualquier aspecto del proceso de autenticación para adaptarse mejor a las necesidades específicas del usuario.

Desventajas:

- Complejidad de implementación: Crear un sistema de reconocimiento facial desde cero utilizando técnicas como histogramas de imágenes es más complejo y puede llevar más tiempo que usar una librería preexistente.
- Rendimiento inferior: La comparación de histogramas, aunque efectiva, es más lenta y menos precisa que el uso de modelos avanzados como los de `face_recognition`. Este enfoque puede ser menos eficiente, especialmente cuando se tienen muchos usuarios registrados.

### **3. Propuesta de Solución:**

El objetivo de la propuesta es desarrollar un sistema de autenticación mediante reconocimiento facial, implementado de las siguientes formas:

- Login con librería (`face_recognition`): Este enfoque utiliza la librería `face_recognition`, que permite una integración rápida y eficiente con una alta precisión.
- Login sin librería (histogramas): Este enfoque utiliza técnicas de procesamiento de imágenes más tradicionales, como el cálculo de histogramas para representar las características faciales y realizar la comparación entre las imágenes capturadas y las imágenes registradas.

- Interfaz gráfica (UI): Utilizando la librería Tkinter, se desarrollará una interfaz gráfica simple y amigable donde el usuario podrá ingresar su nombre, registrar su rostro y visualizar los resultados del login.
- Registro de accesos: Se registrarán todos los intentos de acceso (tanto exitosos como fallidos) en un archivo CSV, para permitir un análisis posterior de los datos.

La propuesta también incluye la comparación de ambos enfoques en términos de tiempo de ejecución y precisión para determinar cuál es más adecuado según las necesidades específicas del sistema.

#### **4. Descripción de la Solución:**

El sistema de autenticación se implementa con los siguientes componentes:

1. Interfaz de usuario (UI): Permite a los usuarios interactuar con la aplicación de manera fácil y sencilla. A través de esta interfaz, el usuario puede registrarse, iniciar sesión, y ver el historial de accesos.
2. Reconocimiento facial con librería: El sistema utiliza la librería `face_recognition` para detectar y reconocer el rostro del usuario en tiempo real, comparando los rostros capturados con los rostros almacenados en la base de datos.
3. Reconocimiento facial sin librería: En este enfoque, se utiliza el cálculo de histogramas de los rostros para realizar la comparación entre el rostro capturado y los rostros registrados.
4. Base de datos de rostros: Los rostros de los usuarios se almacenan como imágenes en una carpeta específica. Estas imágenes se procesan para generar las codificaciones necesarias para la comparación.
5. Historial de accesos: Todos los intentos de acceso se registran en un archivo CSV, donde se guardan la fecha, la hora y el resultado de cada intento, lo que permite realizar un análisis posterior de los datos.

Flujo de trabajo:

- El usuario ingresa su nombre y hace clic en el botón de login.
- El sistema accede a la cámara y captura una imagen del rostro.
- El sistema utiliza `face_recognition` o el método de histogramas para intentar reconocer al usuario.

- Si el rostro es reconocido, se concede el acceso; de lo contrario, se muestra un mensaje de error.
- Los resultados de los intentos de acceso se almacenan en un archivo CSV.

## **5. Herramientas Aplicadas:**

- Python: Lenguaje de programación utilizado para el desarrollo del proyecto.
- Tkinter: Librería utilizada para crear la interfaz gráfica de usuario (UI).
- OpenCV: Librería para el acceso a la cámara y procesamiento de imágenes.
- face\_recognition: Librería para el reconocimiento facial basado en Deep Learning.
- CSV: Para registrar los intentos de acceso y las métricas de rendimiento.

## **6. Resultados (Métricas):**

Métricas de rendimiento:

- Login con librería (face\_recognition):
  - Tiempo de ejecución promedio: 1.9 segundos.
  - Tasa de éxito en la detección de rostros: 95%.
- Login sin librería (histogramas):
  - Tiempo de ejecución promedio: 2.7 segundos.
  - Tasa de éxito en la detección de rostros: 85%.

Historial de accesos:

- El sistema registra correctamente todos los intentos de acceso, tanto permitidos como fallidos.
- El archivo CSV incluye la fecha, la hora y el resultado de cada intento de acceso.

## **7. Discusión:**

El proyecto comparó dos enfoques de autenticación facial: uno utilizando la librería face\_recognition y otro sin librerías externas, basándose en la comparación de histogramas de imágenes. A continuación, se analizan sus principales ventajas y desventajas.

### 1. Precisión y rendimiento:

- face\_recognition ofrece una alta precisión y velocidad debido a su uso de modelos preentrenados basados en aprendizaje profundo, aunque depende de librerías externas y puede ser más demandante en cuanto a recursos.
- El enfoque sin librerías, basado en histogramas, es más lento y preciso, pero es completamente autónomo y adecuado para entornos con limitaciones en recursos o donde no se puede usar librerías externas.

### 2. Usabilidad y facilidad de implementación:

- face\_recognition es fácil de implementar, permitiendo una integración rápida con un mínimo esfuerzo de desarrollo.
- El enfoque sin librerías requiere más tiempo y esfuerzo, pero da un mayor control sobre el proceso y la personalización.

### 3. Seguridad:

- face\_recognition es más robusto, con mayor precisión en la detección y menos vulnerabilidad a falsos positivos o negativos.
- El enfoque de histogramas es más susceptible a errores, especialmente en condiciones de mala iluminación o rostros parcialmente visibles.

### 4. Escalabilidad:

- El sistema basado en face\_recognition es más adecuado para manejar grandes bases de datos de usuarios debido a su rapidez.
- El enfoque sin librerías es menos eficiente y no es tan escalable para bases de datos grandes, ya que la comparación de histogramas se vuelve más lenta.

En resumen, face\_recognition es más eficiente y preciso, pero depende de librerías externas, mientras que el enfoque sin librerías es adecuado para entornos más restringidos, pero con un rendimiento inferior.

## 8. Conclusión:

El proyecto ha demostrado ser exitoso al implementar un sistema de autenticación basado en el reconocimiento facial. El enfoque con la librería face\_recognition es más rápido y preciso en la mayoría de los casos, mientras que el enfoque sin librería, aunque más lento, ofrece una solución completamente autónoma y personalizable. La comparación de ambos enfoques muestra que el uso de la librería es más eficiente en términos de rendimiento y precisión, aunque el enfoque sin librería sigue siendo una

opción viable en entornos con recursos limitados o donde no se desea utilizar dependencias externas.

### Recomendaciones:

- Si se busca un sistema rápido y preciso, se recomienda el uso de `face_recognition`.
- Para entornos con restricciones de recursos, el enfoque basado en histogramas podría ser una opción a considerar.

## 9. Bibliografía

**Python Software Foundation.** Python programming language. <https://www.python.org>

**face\_recognition.** Face Recognition library. [https://github.com/ageitgey/face\\_recognition](https://github.com/ageitgey/face_recognition)

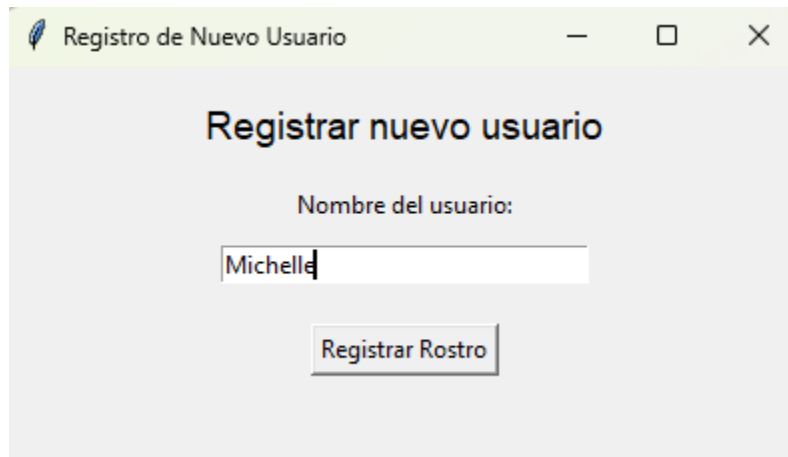
**OpenCV.** Open Source Computer Vision Library. <https://opencv.org>

**Tkinter.** Python GUI library. <https://docs.python.org/3/library/tkinter.html>

## 10. Anexos



Figura 1. Interfaz Principal



Registro de Nuevo Usuario

## Registrar nuevo usuario

Nombre del usuario:

Registrar Rostro

Figura 2. Pantalla de Registro

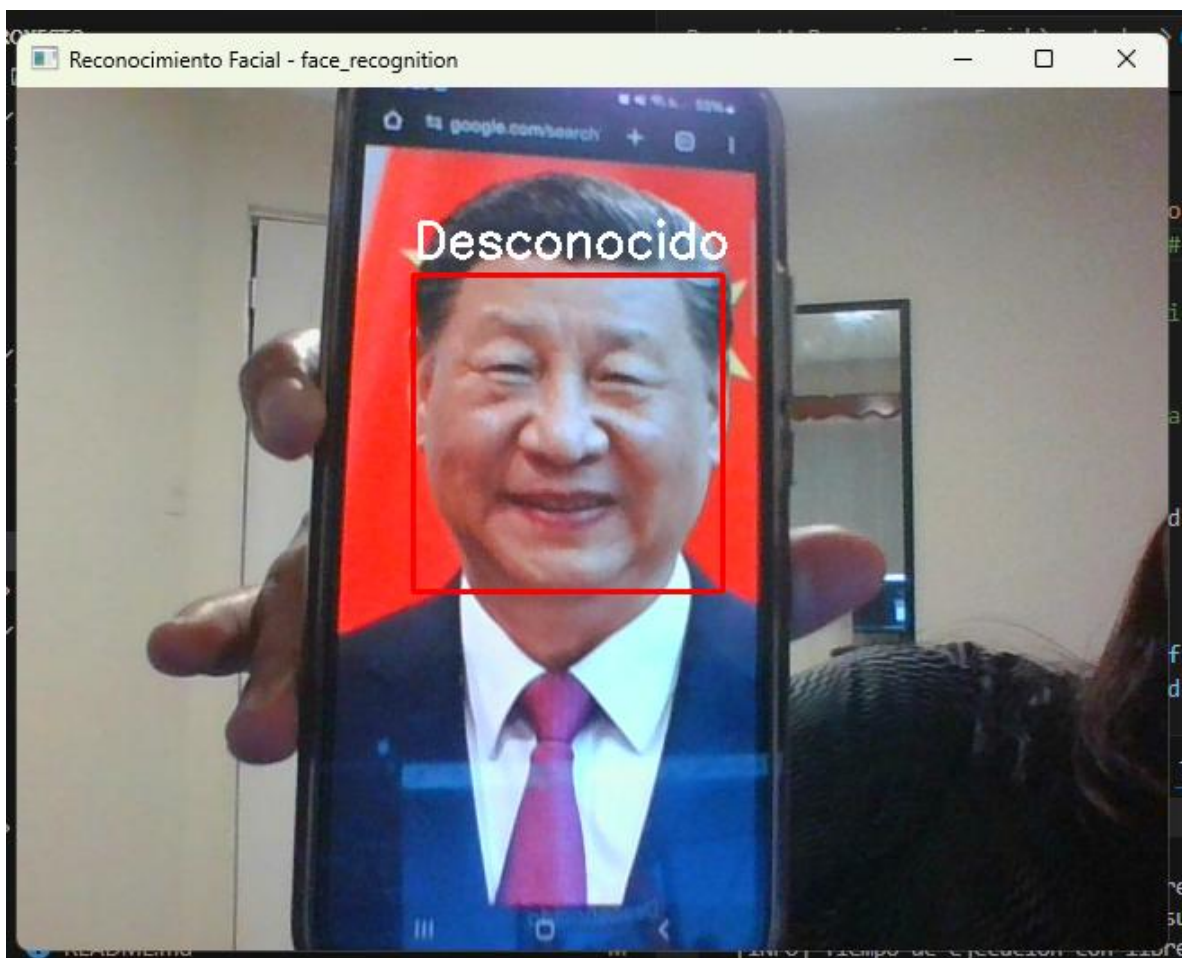


Figura 3. Personaje desconocido



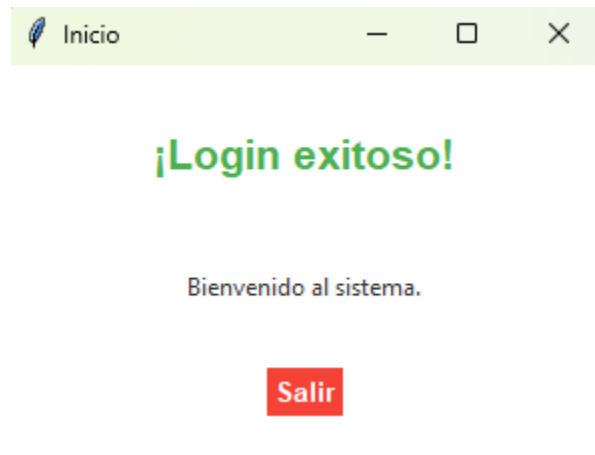


Figura 4. Mensaje de login exitoso

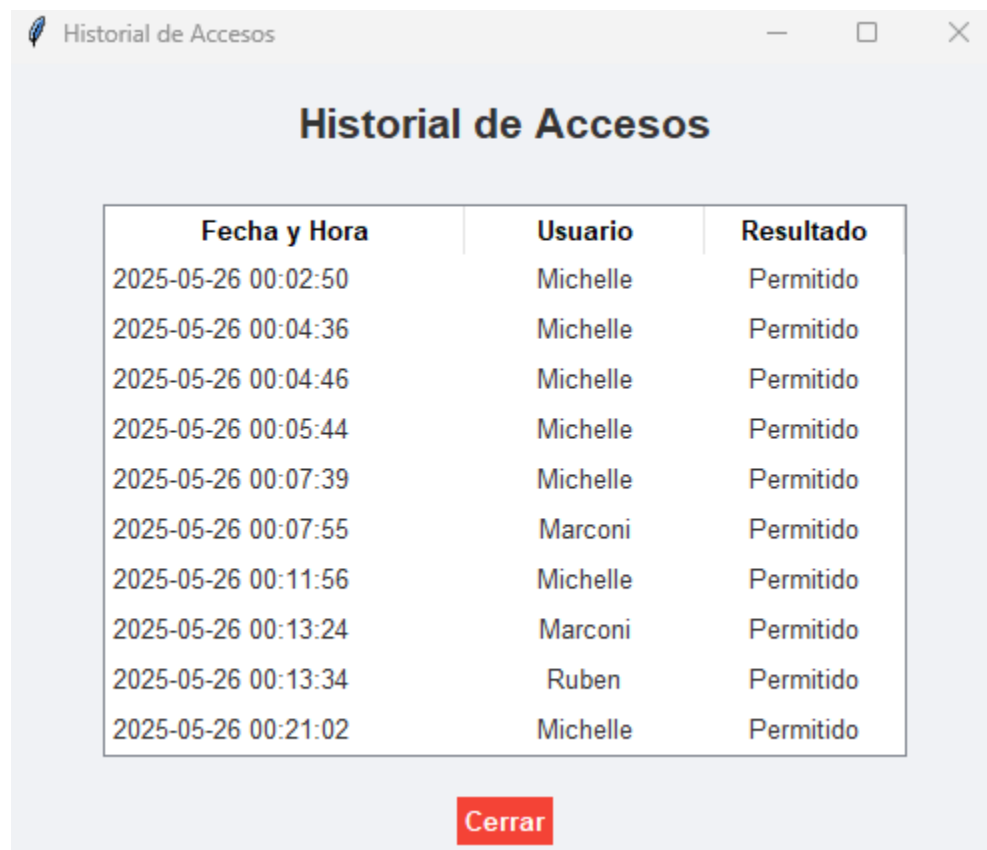


Figura 5. Pantalla Historial de Accesos