

Manual Técnico del Sistema de Reconocimiento Facial con Flask

Versión 1.6

Fecha: Agosto 2025

Autor: Equipo de Desarrollo

1. Introducción

Este manual técnico describe la arquitectura, componentes, instalación, configuración, operación y mantenimiento del sistema de reconocimiento facial desarrollado con Flask, OpenCV y face_recognition. El objetivo es proporcionar a administradores y desarrolladores la información necesaria para desplegar, operar y extender la aplicación.

2. Descripción General del Sistema

El sistema permite la detección y reconocimiento de rostros en tiempo real mediante una cámara ESP32-CAM. Integra una interfaz web para gestionar usuarios, personas autorizadas y revisar registros de intrusos. Utiliza embeddings faciales para la verificación de identidad y almacena datos en SQLite y el sistema de archivos.

3. Arquitectura

La arquitectura sigue una separación por capas:

- Capa Web (Flask): autenticación, sesiones, vistas HTML, API JSON.
- Motor de Reconocimiento: procesamiento de video, detección y comparación de rostros.
- Persistencia: base de datos SQLite y sistema de archivos para imágenes y embeddings.
- Dispositivo de captura: ESP32-CAM como fuente de video MJPEG.

4. Requisitos

4.1 Hardware

- Servidor o PC con Python 3.8 o superior.
- ESP32-CAM con firmware para transmitir video MJPEG.
- Cámara con resolución mínima de 640x480.
- Almacenamiento suficiente para imágenes y base de datos.

4.2 Software

- Sistema operativo: Linux, Windows o macOS.
- Python 3.8+.
- Dependencias listadas en requirements.txt.
- Navegador web compatible (Chrome, Firefox, Edge).

5. Instalación

- 1) Clonar el repositorio o copiar los archivos del proyecto.
- 2) Crear y activar un entorno virtual de Python.
- 3) Instalar las dependencias:

```
pip install -r requirements.txt
```
- 4) Configurar variables de entorno (SECRET_KEY, ESP32_STREAM_URL).
- 5) Crear las carpetas data/autorizados y data/intrusos (si no existen).
- 6) Ejecutar la aplicación con:

```
python app.py
```

6. Configuración

- SECRET_KEY: clave para sesiones Flask.
- ESP32_STREAM_URL: URL del stream de la cámara ESP32-CAM.
- AUTH_TOLERANCE e INTRUSO_TOLERANCE: tolerancias de comparación facial.
- Debug: desactivar en producción.

7. Operación

- 1) Acceder vía navegador a la dirección del servidor.
- 2) Iniciar sesión con credenciales válidas.
- 3) Para transmitir video: ir a la sección de transmisión.
- 4) Para registrar autorizados: cargar imagen o capturar desde la cámara.
- 5) Revisar intrusos en su sección y eliminarlos si es necesario.
- 6) Administrar usuarios (solo admin).

8. Mantenimiento

- Respalidar regularmente la base de datos y las carpetas de datos.
- Revisar y rotar la SECRET_KEY periódicamente.
- Actualizar dependencias de Python con precaución.
- Monitorear el uso de disco y limpiar imágenes obsoletas.

9. Solución de Problemas

- Error de conexión a cámara: verificar ESP32-CAM y URL.
- Reconocimiento lento: reducir resolución o ajustar tolerancias.
- Fallos en instalación: revisar versión de Python y dependencias.

10. Seguridad

- Cambiar credenciales por defecto inmediatamente.
- Usar HTTPS para proteger la transmisión.
- Restringir acceso a la interfaz de administración.
- Implementar protección CSRF para formularios.

11. Anexos

- requirements.txt con dependencias.
- Scripts de inicialización de la base de datos.
- Estructura de carpetas del proyecto.