Universidad Nacional de Ingeniería
Facultad de Ciencias

Asistente Virtual "LemurAI"

Integrantes	Código
Pacheco Taboada André Joaquín	20222189G
Perez Villegas, Arbues Enrique	20220419E
Orihuela Contreras, Jared	20220370F
Quispe Olaechea, Pablo Alejandro	20194153G

Asociación Científica Especializada en Computación

Mayo 2024

Índice

1. Introducción **3**

2. Materiales utilizados **3**
2.1. Software 3

3. Metodología **5**
3.1. Análisis de Requerimientos 6
3.2. Diseño del Sistema 6
3.3. Implementación 6
3.4. Desarrollo 7

4. Resultados obtenidos **7**
4.1. Clasificación de Caras 7
4.2. Respuesta del LLM 8
4.3. Generación de Audio 8

5. Conclusiones **9**

1. Introducción

Este proyecto fue concebido con el objetivo de mejorar la interacción tecnológica dentro de la asociación Acecom, utilizando tecnologías avanzadas de reconocimiento facial y asistencia por voz. El reconocimiento facial se emplea para identificar a los miembros al ingresar al local, facilitando un proceso de registro automático. Paralelamente, un asistente de voz mejora la experiencia del usuario al proporcionar respuestas personalizadas mediante el uso de tecnología de procesamiento de lenguaje natural y síntesis de voz.

La integración de múltiples tecnologías y plataformas, como OpenCV para el procesamiento de imágenes, Keras y Facenet para el reconocimiento facial, y un modelo de lenguaje grande (LLM) junto con tecnología de texto a voz (TTS), permite no solo la identificación de los miembros sino también la interacción en tiempo real con ellos de una manera más natural y personalizada.

En el diagrama adjunto se observa el flujo de trabajo del sistema. La cámara detecta y reconoce los rostros, enviando la información a un módulo de control de flujo que registra la aparición de personas y activa otras funcionalidades según sea necesario. El micrófono capta el audio, que es transformado en texto por el módulo AudioToText, y este texto es procesado por el LLM para generar respuestas relevantes y personalizadas. Posteriormente, el texto es convertido nuevamente en audio por el Modulador de Voz para ser emitido por el parlante. El Orquestador coordina todas estas operaciones y mantiene un registro histórico de las interacciones.

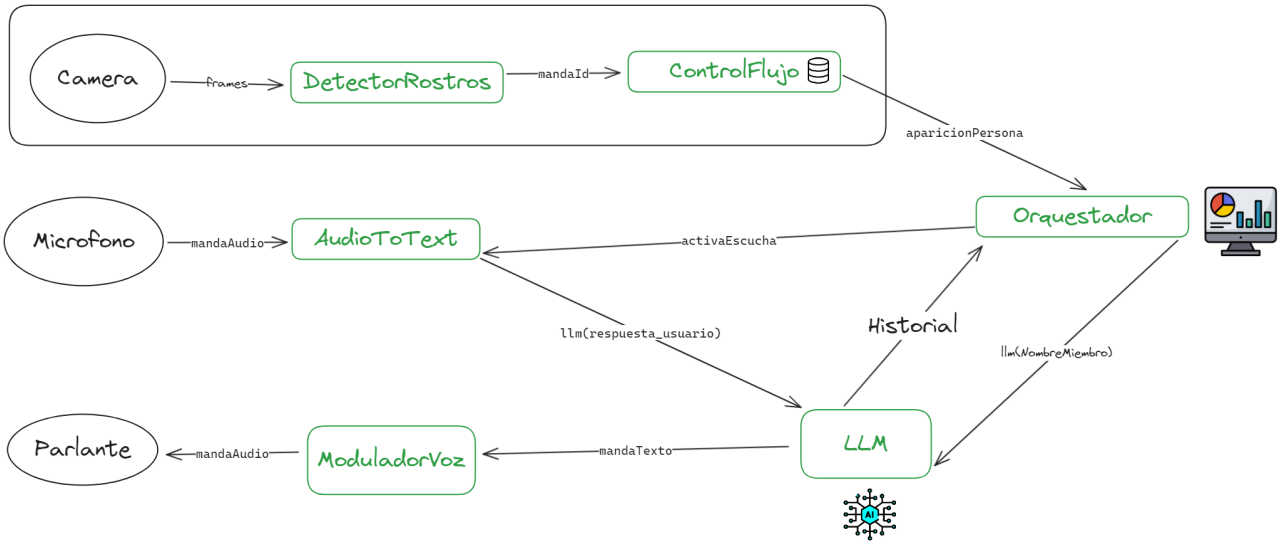


Figura 1: Diagrama del sistema de reconocimiento facial y asistencia por voz implementado en Acecom.

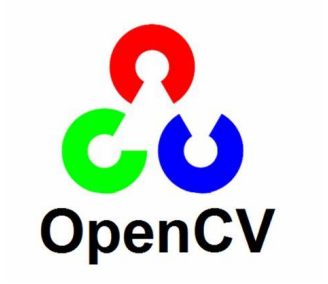
A continuación, detallamos los materiales utilizados, la metodología empleada, los resultados obtenidos y las conclusiones derivadas de nuestro trabajo. Este informe ofrece una visión comprensiva del proyecto, destacando las contribuciones tecnológicas y los beneficios derivados para Acecom.

2. Materiales utilizados

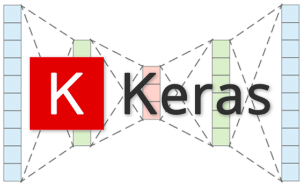
2.1. Software

- **Python:** Lenguaje de programación principal utilizado para la implementación del reconocimiento facial, integrando diversas librerías especializadas.

- **OpenCV:** Biblioteca de visión por computadora usada para el procesamiento de las imágenes y detección de rostros.

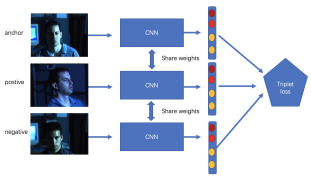


- **Keras:** Libreria que nos ayudara para la extraccion de modelos ya entrenados como facenet



- **scikit-learn:** Librería para aprendizaje de máquina que proporciona herramientas para la selección de modelos y la evaluación del clasificador.

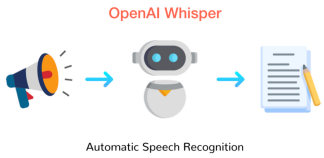
- **Facenet:** Red neuronal preentrenada utilizada para la extracción de características faciales y la clasificación de identidades.



- **LLM (Llama3 8b):** Modelo de lenguaje de gran escala utilizado para generar respuestas personalizadas en texto, capaz de saludar a los usuarios por su nombre.

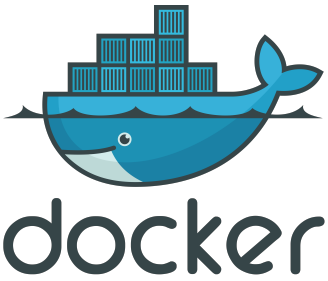
- **Langchain:** Framework que abstrae las complejidades que conlleva interactuar con los LLM y proporciona herramientas para incorporar eficazmente las capacidades del LLM al proyecto.

- **WhisperX:** Tecnología de reconocimiento de voz que convierte audio a texto (Speech-to-Text) con alta precisión. Utilizada para mejorar la interacción de los usuarios al permitirles comunicarse verbalmente con el sistema.



- **TTS (Text-to-Speech) con modelo VITS:** Tecnología de síntesis de voz que convierte el texto en habla, utilizado para la interacción vocal en español.
- **PostgreSQL:** Sistema de gestión de bases de datos relacionales (RDBMS) encargado de lidiar con la base de datos de registros de entrada y salida de miembros del local.

- **Docker:** Plataforma utilizada para la contenerización de las aplicaciones, facilitando su despliegue y escalabilidad.



- **Kubernetes:** Sistema de orquestación de contenedores que gestiona la automatización del despliegue, escalado y operaciones de las aplicaciones contenerizadas.

3. Metodología

La metodología seguida para el desarrollo del proyecto de reconocimiento facial y asistente de voz en Acecom se estructura en varias fases, detallando cada actividad crucial para el éxito del sistema.

A continuación, se describen las etapas del proceso:

3.1. Análisis de Requerimientos

Para alcanzar los objetivos propuestos de automatización del reconocimiento facial de los miembros y facilitar su registro automático en la base de datos, además de brindar un asistente de voz al local de Acecom, se realizaron las siguientes actividades:

- **Identificación de Stakeholders:** Se identificaron los principales usuarios y beneficiarios del sistema, incluyendo miembros de Acecom, para entender sus necesidades y expectativas.
- **Levantamiento de Requerimientos Funcionales:** Se definieron las capacidades esenciales del sistema, tales como la detección y reconocimiento facial en tiempo real, el registro en base de datos y la generación de respuestas personalizadas a través de un asistente de voz.
- **Requerimientos No Funcionales:** Se establecieron criterios para asegurar la privacidad y seguridad de los datos, la escalabilidad del sistema y la capacidad de integrarse fácilmente con tecnologías existentes y futuras.
- **Evaluación de Tecnologías:** Se seleccionaron las tecnologías más adecuadas para implementar las funcionalidades requeridas, como Python, OpenCV, Keras, scikit-learn, Docker y Kubernetes.

3.2. Diseño del Sistema

El sistema se diseñó para ser modular, escalable y fácil de mantener, con los siguientes componentes clave:

- **Módulo de Reconocimiento Facial:** Utilizando Facenet junto con OpenCV y Keras para la detección y reconocimiento facial preciso. Este módulo procesa imágenes de cámaras en tiempo real para identificar a los miembros.
- **Base de Datos:** Se implementó una base de datos para almacenar la información de los rostros reconocidos, además de la hora y fecha de registro de cada miembro.
- **Asistente de Voz:** Integración de Llama3 8b con Langchain para generar respuestas personalizadas y el uso de la librería TTS con el modelo VITS en español para la síntesis de voz, permitiendo interactuar con los usuarios mediante mensajes verbales.
- **Contenerización y Orquestación:** Cada componente del sistema fue contenerizado utilizando Docker, y la gestión de los contenedores se realiza a través de Kubernetes, lo que facilita la escalabilidad y la gestión del despliegue.

3.3. Implementación

- **Desarrollo de Componentes**: Implementar los módulos individuales utilizando las librerías y frameworks seleccionados. Esto incluye la configuración del reconocimiento facial con Facenet y OpenCV, la generación de respuestas con Llama3 8b, y la síntesis de voz con el modelo VITS.
- **Contenerización de Servicios**: Utilizar Docker para crear imágenes de contenedores para cada parte del sistema, asegurando su correcta ejecución en diferentes entornos.
- **Orquestación con Kubernetes**: Configurar Kubernetes para manejar el despliegue, la escalabilidad y la gestión de los contenedores, facilitando el mantenimiento y la expansión del sistema.

- **Desarrollo de Componentes**: Implementar los módulos individuales utilizando las librerías y frameworks seleccionados. Esto incluye la configuración del reconocimiento facial con Facenet y OpenCV, la generación de respuestas con Llama3 8b, y la síntesis de voz con el modelo VITS.
- **Contenerización de Servicios**: Utilizar Docker para crear imágenes de contenedores para cada parte del sistema, asegurando su correcta ejecución en diferentes entornos.
- **Orquestación con Kubernetes**: Configurar Kubernetes para manejar el despliegue, la escalabilidad y la gestión de los contenedores, facilitando el mantenimiento y la expansión del sistema.

3.4. Desarrollo

El desarrollo del sistema se centró en la creación de software robusto y eficiente, utilizando metodologías ágiles y prácticas de programación de alto nivel:

- **Programación de Módulos:** Cada módulo fue desarrollado en Python, utilizando librerías como OpenCV, Keras y scikit-learn para el reconocimiento facial y Facenet para la extracción y clasificación de características faciales.
- **Integración de LLM y TTS:** Se implementaron APIs para conectar el sistema de reconocimiento facial con el modelo de lenguaje Llama3 8b y la síntesis de voz VITS, permitiendo interacciones fluidas y naturales en español.
- **Pruebas Unitarias y de Integración:** Se llevaron a cabo pruebas unitarias para cada módulo y pruebas de integración para asegurar la compatibilidad y el rendimiento del sistema completo.
- **Control de Versiones:** Se utilizó Git para el control de versiones, permitiendo la colaboración entre los desarrolladores y la gestión de cambios de manera eficaz.
- **Documentación:** Se elaboró documentación técnica detallada para cada componente del sistema, incluyendo especificaciones de la API, manuales de usuario y guías de instalación.

4. Resultados obtenidos

Los resultados obtenidos de la implementación y operación del sistema en Acecom demuestran la eficacia y funcionalidad del sistema de reconocimiento facial y asistencia por voz. A continuación, se presentan algunas evidencias clave de su rendimiento:

4.1. Clasificación de Caras

La funcionalidad de reconocimiento facial del sistema mostró una alta precisión en la identificación de los miembros de Acecom, incluso en condiciones de iluminación variada y diferentes ángulos de cámara. A continuación se muestra un ejemplo de los resultados de la clasificación de caras:

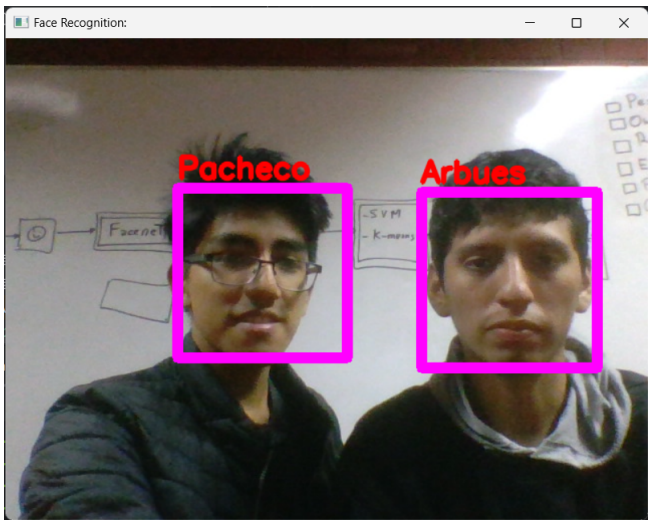


Figura 2: Ejemplo de resultado de clasificación facial.

4.2. Respuesta del LLM

El modelo de lenguaje Llama3 8b, integrado con Langchain, fue capaz de generar respuestas personalizadas basadas en el reconocimiento facial. Este aspecto del sistema mejoró significativamente la interacción con los usuarios, haciendo que el asistente de voz fuera más útil y agradable. Se muestra a continuación un ejemplo de la respuesta generada por el LLM:

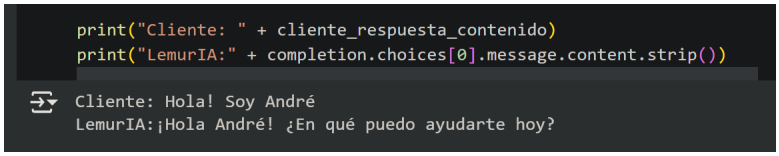


Figura 3: Ejemplo de respuesta personalizada del LLM al usuario.

4.3. Generación de Audio

La síntesis de voz utilizando el modelo TTS VITS en español permitió una comunicación clara y natural. El audio generado fue coherente con el texto proporcionado por el LLM, ofreciendo una experiencia de usuario fluida y comprensible. A continuación se muestra una representación visual del audio generado:

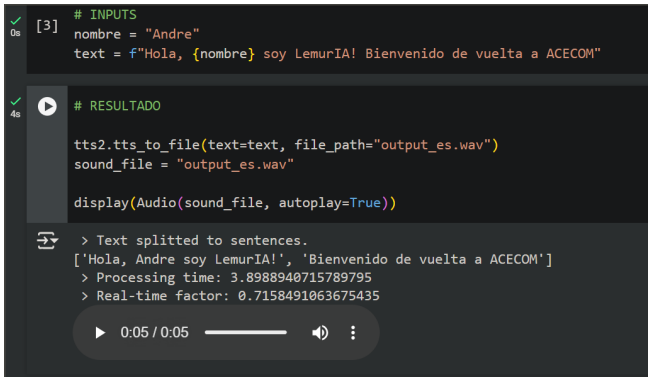


Figura 4: Visualización del audio generado por el TTS.

Estos resultados validan la efectividad del sistema en proporcionar una solución integrada de reconocimiento facial y asistencia por voz, destacando su capacidad para mejorar la interacción entre los miembros de Acecom y la tecnología implementada.

5. Conclusiones

Este proyecto ha logrado con éxito sus objetivos de automatizar el reconocimiento facial de los miembros de Acecom y de facilitar su registro automático en la base de datos, así como de brindar un asistente de voz eficaz en el local. A continuación, se resumen los puntos clave y los aprendizajes del proyecto:

- **Eficiencia en el Reconocimiento Facial:** La integración de Facenet con OpenCV y Keras junto con una arquitectura MTCNN para el preprocesamiento de los datos y un clasificador como K-means o SVM ha demostrado ser altamente eficaz para la identificación precisa de los miembros bajo diversas condiciones ambientales. Esto ha mejorado significativamente la seguridad y la gestión de accesos en el local.
- **Interacción Enriquecida mediante Asistente de Voz:** La implementación del modelo Llama3 8b y del sistema TTS VITS ha permitido crear un asistente de voz que no solo reconoce a los miembros por su nombre, sino que también facilita una interacción natural y personalizada, mejorando la experiencia del usuario en el local de Acecom.
- **Integración Tecnológica Exitosa:** La armonización de múltiples tecnologías, desde el reconocimiento facial hasta la síntesis de voz, pasando por la gestión de bases de datos y la orquestación de contenedores, ilustra la capacidad del equipo para abordar y superar complejidades técnicas significativas.
- **Escalabilidad y Mantenimiento:** La contenerización con Docker y la orquestación con Kubernetes han asegurado que el sistema no solo sea escalable, sino también mantenible con una inversión de tiempo y recursos relativamente baja, proyectando un soporte sostenible a largo plazo.
- **Retroalimentación y Mejoras Continuas:** La interacción con los usuarios finales y su retroalimentación han sido cruciales para iterar y mejorar continuamente el sistema. Este enfoque ha asegurado que el sistema no solo cumpla con los requisitos técnicos, sino que también satisfaga las necesidades reales de los usuarios.

En conclusión, el proyecto no solo ha alcanzado sus metas propuestas, sino que también ha sentado las bases para futuras expansiones y mejoras. La infraestructura implementada ofrece un marco robusto para integrar nuevas funcionalidades, como la ampliación de la base de datos para incluir más puntos de datos de los miembros o la introducción de nuevas capacidades de inteligencia artificial para análisis predictivo de necesidades de los usuarios. Estas direcciones no solo expandirán la funcionalidad del sistema sino que también fortalecerán el papel de la tecnología en mejorar la interacción humana en entornos organizacionales.