UNIVERSIDAD EUROPEA DE MADRID

ESCUELA DE ARQUITECTURA, INGENIERÍA Y DISEÑO

MÁSTER UNIVERSITARIO EN ANALISIS DE DATOS MASIVOS (BIG DATA)

TRABAJO FIN DE MÁSTER

Desarrollo y despliegue de un sistema de búsqueda de rostros a partir de descripciones en lenguaje natural.

Omaira González Pérez
Dirigido por
Miguel Torres Porta
CURSO 2024-2025

TÍTULO: Desarrollo y despliegue de un sistema de búsqueda de rostros a partir de descripciones en lenguaje natural.

AUTOR: Omaira González Pérez

TITULACIÓN: MÁSTER UNIVERSITARIO EN ANALISIS DE DATOS MASIVOS (BIG DATA)

DIRECTOR/ES DEL PROYECTO: Miguel Torres Porta

FECHA: septiembre de 2025

RESUMEN

Este proyecto surge para resolver la dificultad de buscar personas en grandes colecciones de imágenes a partir de descripciones escritas, un reto habitual en sistemas de gestión visual y análisis de datos. Para ello se diseñó y desarrolló un sistema que combina generación automática de descripciones de imágenes con búsqueda semántica mediante inteligencia artificial. Se utilizaron modelos BLIP para convertir imágenes en texto y CLIP para medir la similitud entre descripciones e imágenes, logrando identificar de manera eficiente los resultados más relevantes para cada consulta.

La solución se desplegó en el proveedor de servicios en la nube Azure, con el frontend desplegado en Static Web Apps e integrado con GitHub, lo que permite actualizar automáticamente la interfaz con cada cambio en el repositorio. El backend corre en una máquina virtual gestionada con Nginx y Gunicorn, garantizando seguridad, escalabilidad y accesibilidad desde cualquier red.

Aunque el proyecto se desarrolló de forma independiente, sus resultados pueden ser útiles para sectores como seguridad, análisis de medios o gestión de bases de datos visuales. Entre los logros más importantes se encuentran la capacidad de generar descripciones coherentes, organizar imágenes según su relevancia semántica y ofrecer un sistema funcional y confiable en tiempo real. En conclusión, el proyecto demuestra que es posible combinar inteligencia artificial, despliegue en la nube y automatización de repositorios para crear soluciones prácticas, escalables y mantenibles en el ámbito de la visión artificial y la recuperación semántica de información.

Palabras clave: visión artificial, inteligencia artificial, búsqueda semántica, Azure, Static Web Apps, GitHub.

ABSTRACT

This project addresses the challenge of searching for people in large image collections based on written descriptions, a common issue in visual management and data analysis systems. To solve this, a system was designed and developed that combines automatic image captioning with semantic search using artificial intelligence. BLIP models were used to convert images into text, and CLIP models to measure the similarity between descriptions and images, efficiently identifying the most relevant results for each query.

The solution was deployed in the cloud using Azure, with the frontend hosted on Static Web Apps and integrated with GitHub, allowing automatic updates whenever changes are made to the repository. The backend runs on a virtual machine managed with Nginx and Gunicorn, ensuring security, scalability, and accessibility from any network.

Although the project was developed independently, its outcomes could be valuable for sectors such as security, media analysis, or visual database management. Key achievements include generating coherent descriptions, ranking images by semantic relevance, and providing a functional, reliable system in real time. In conclusion, the project demonstrates that combining artificial intelligence, cloud deployment, and repository automation allows the creation of practical, scalable, and maintainable solutions for visual recognition and semantic information retrieval.

Keywords: computer vision, artificial intelligence, semantic search, Azure, Static Web Apps, GitHub.

TABLA RESUMEN

	DATOS
Nombre y apellidos:	Omaira González Pérez
Título del proyecto:	Desarrollo y despliegue de un sistema de búsqueda de rostros a partir de descripciones en lenguaje natural.
Directores del proyecto:	Miguel Torres Porta
El proyecto se ha realizado en colaboración de una empresa o a petición de una empresa:	NO
El proyecto ha implementado un producto: (esta entrada se puede marcar junto a la siguiente)	SI
El proyecto ha consistido en el desarrollo de una investigación o innovación: (esta entrada se puede marcar junto a la anterior)	NO
Objetivo general del proyecto:	Facilitar la búsqueda de personas a partir de una descripción escrita mediante un sistema inteligente de reconocimiento y recuperación de imágenes

Índice

Índice de Figuras

Capítulo 1. RESUMEN DEL PROYECTO

En este capítulo se presentan, de forma breve y clara, los ejes principales del Trabajo Fin de Máster

1.1 Contexto y justificación

En muchos escenarios (testimonios, avisos en tiempo real o consultas con información incompleta) no se dispone de una fotografía, pero sí de una descripción verbal de la persona buscada. Las soluciones actuales de reconocimiento facial suelen requerir imágenes de referencia o condiciones muy controladas, por lo que su utilidad se limita en estos casos. Al mismo tiempo, han surgido avances en IA que acercan texto e imagen: por un lado, modelos de lenguaje capaces de entender descripciones complejas; por otro, modelos visión-lenguaje que relacionan esas descripciones con rostros en bases de imágenes. Su combinación permite buscar, a partir de texto, qué rostros encajan mejor, con especial interés en seguridad y análisis forense, y utilidad potencial en otros ámbitos donde se filtra o recomienda por rasgos descritos. Este planteamiento exige, además, atender de forma explícita a la privacidad, los sesgos y la transparencia del sistema, en consonancia con el marco regulatorio vigente.

1.2 Planteamiento del problema

En escenarios donde solo se dispone de una descripción verbal, no existe una forma fiable de localizar un rostro en grandes bases de imágenes sin una foto previa. De aquí nace la pregunta motriz: ¿cómo convertir una descripción en una búsqueda operativa, precisa y ágil sobre un repositorio de rostros, cumpliendo además con la normativa y las garantías éticas? El proyecto se aborda como desarrollo y despliegue de una solución funcional. Se plantea el desarrollo de una herramienta práctica orientada a uso real, una herramienta funcional que transforme descripciones en búsquedas sobre bases de imágenes.

En los últimos años, la inteligencia artificial ha experimentado avances notables en áreas como la comprensión del lenguaje y el análisis de imágenes. Esta evolución ha propiciado el desarrollo de sistemas capaces de procesar grandes volúmenes de información textual y visual, dando lugar a nuevas formas de interacción entre humanos y máquinas. Sin embargo, pese al crecimiento acelerado en este ámbito, hay una carencia importante en cuanto a la capacidad de los sistemas automatizados para vincular descripciones en lenguaje natural con rostros humanos presentes en bases de datos visuales.

Actualmente, muchas de las herramientas disponibles para el reconocimiento de personas a través de sistemas informáticos requieren imágenes de referencia, lo que limita significativamente su utilidad en escenarios donde solo se dispone de una descripción verbal. Esto ocurre, por ejemplo, en contextos forenses, sistemas de seguridad, análisis de testimonios, o incluso en aplicaciones comerciales donde se desea filtrar o recomendar perfiles basándose únicamente en características descritas por el usuario. Aunque existen modelos potentes como CLIP o BLIP-2, capaces de relacionar texto e imagen, y modelos de lenguaje como GPT-4 que interpretan descripciones complejas con gran precisión, la integración de

estas capacidades en una solución funcional y específica para la búsqueda facial basada en texto sigue siendo un reto.

Aunque ya tiene una base bastante sólida, todavía no existe una herramienta realmente práctica y eficaz que transforme una descripción en lenguaje natural en una búsqueda útil dentro de un sistema de reconocimiento facial. Para que eso funcione bien, no solo hace falta interpretar correctamente lo que se dice, sino también contar con una forma de representar tanto el texto como las imágenes en un mismo "lenguaje" o espacio común, de modo que puedan compararse de manera lógica y con sentido.

Esta problemática pone de manifiesto una necesidad concreta en el campo de la inteligencia artificial aplicada: diseñar un sistema capaz de interpretar descripciones humanas, transformarlas en representaciones computacionales adecuadas y compararlas con una base de datos de imágenes para obtener coincidencias visuales relevantes. La solución a este problema implicaría combinar técnicas de procesamiento del lenguaje natural, procesamiento de imágenes y aprendizaje multimodal, integrando modelos previamente entrenados y optimizando su uso para un objetivo concreto. En definitiva, se trata de cubrir una laguna existente entre el potencial de los modelos actuales y su aplicación efectiva en sistemas orientados a tareas de identificación visual a partir del lenguaje.

En este contexto, el despliegue de la solución en una plataforma en la nube como Microsoft Azure ofrece un entorno adecuado para dar respuesta al problema. Azure permite integrar modelos de visión y lenguaje de manera escalable, facilitando tanto el procesamiento intensivo de imágenes como la gestión segura de datos sensibles. Su infraestructura distribuida hace posible ejecutar cargas de trabajo en tiempo real y mantener la disponibilidad del sistema, garantizando que la herramienta pueda crecer y adaptarse a nuevas necesidades sin perder fiabilidad ni rendimiento.

1.3 Objetivos del proyecto

El objetivo de esto proyecto es construir un sistema que interprete descripciones en lenguaje natural y las compare con una base de imágenes de rostros para identificar el perfil más similar, combinando técnicas de procesamiento de lenguaje natural, visión por computador y modelos multimodales.

1.4 Resultados obtenidos

Al finalizar el proyecto, los resultados obtenidos muestran que el sistema es capaz de generar descripciones precisas de las imágenes mediante el modelo BLIP y realizar búsquedas semánticas eficientes usando CLIP, ofreciendo un ranking de imágenes ordenadas según la similitud con la descripción de consulta. Las evaluaciones realizadas indican que, aunque algunas descripciones generadas son genéricas, el sistema consigue identificar correctamente imágenes visualmente coherentes con la consulta, demostrando la efectividad del enfoque basado en inteligencia artificial para la recuperación de imágenes. Además, el despliegue en

Azure garantiza que la aplicación sea accesible desde cualquier red, segura y confiable, mientras que la integración con GitHub permite mantener el frontend actualizado automáticamente, asegurando un flujo de trabajo ágil y consistente. Estos resultados confirman que la solución cumple los objetivos planteados, combinando precisión en la búsqueda con un despliegue eficiente y escalable.

1.5 Estructura de la memoria

La memoria se estructura en varios capítulos: el Capítulo 2 revisa los antecedentes y el estado del arte, el Capítulo 3 presenta los objetivos generales y específicos. El Capítulo 4 describe en detalle el desarrollo del proyecto: planificación, construcción de la base de datos, metodologías y herramientas empleadas, despliegue de la solución como aplicación web y una comparación con sistemas actuales que motiva la propuesta alternativa. El Capítulo 5 presenta las conclusiones del trabajo y las personales y el Capítulo 6 las futuras líneas de trabajo.

Capítulo 2. ANTECEDENTES / ESTADO DEL ARTE

En este capítulo se introduce el contexto general en el que se enmarca este trabajo, centrado en el desarrollo de una herramienta capaz de localizar rostros humanos a partir de descripciones escritas. Para ello, se revisan los fundamentos necesarios y las distintas aproximaciones tecnológicas que hacen posible esta tarea, con el fin de ofrecer una comprensión sólida de las opciones exploradas y de los criterios seguidos a lo largo del proceso de diseño y desarrollo.

2.1 Estado del arte

Hay situaciones en las que no se dispone de una imagen, pero sí de una descripción verbal de una persona, y contar con una herramienta capaz de comparar esa información con una base de datos visual sería de gran utilidad, aunque actualmente no exista una solución que resuelva este problema de forma directa y eficaz.

2.2 Contexto y justificación

La inteligencia artificial está transformando rápidamente el sector de la seguridad, especialmente a través del uso de modelos que combinan procesamiento visual y lingüístico. En situaciones donde no se dispone de una imagen del sospechoso o persona buscada (como puede ocurrir en testimonios o reportes en tiempo real), contar con sistemas que puedan operar a partir de descripciones textuales supone una gran ventaja.

Aquí es donde cobran especial relevancia dos avances recientes en el campo de la IA: los **Large Language Models (LLMs)** y los **Visual Language Models (VLMs)**.

Los **LLMs**, como GPT o BERT, han demostrado una capacidad sobresaliente para entender y generar lenguaje natural con un alto nivel de coherencia y contexto. Estos modelos pueden interpretar descripciones complejas y extraer de ellas atributos específicos.

Por otro lado, los **VLMs** permiten conectar ese entendimiento textual con elementos visuales. Modelos como BLIP o CLIP son capaces de analizar imágenes y vincularlas con descripciones escritas, logrando identificar, por ejemplo, qué rostro en una fotografía se ajusta mejor a una descripción dada.

La combinación de ambos tipos de modelos abre la puerta a una nueva generación de agentes inteligentes: sistemas capaces de entender descripciones humanas y buscar rostros en imágenes sin requerir una foto previa del individuo. Esto representa una evolución significativa respecto a los sistemas tradicionales de reconocimiento facial, que suelen requerir imágenes base o funcionar en condiciones muy controladas.

En el contexto de la seguridad, esta capacidad puede ser decisiva para acelerar investigaciones, mejorar la vigilancia en tiempo real o actuar con rapidez en la localización de personas, incluso con información parcial.

Este trabajo se justifica en la necesidad de adaptar las herramientas actuales a los desafíos reales de seguridad, y propone aprovechar el potencial de los LLMs y VLMs para desarrollar un agente capaz de unir texto e imagen en un entorno operativo.

2.2.1 Marco teórico.

Para que los modelos que combinan texto e imágenes, como CLIP o BLIP, puedan realmente "entender" ambos tipos de información, necesitamos una forma de representar el contenido de manera que sea comparable. Esto se debe a que las imágenes y los textos son datos muy distintos: mientras que un texto está formado por palabras y reglas gramaticales, una imagen contiene píxeles, colores, formas y patrones visuales. Para que un modelo pueda relacionar ambos, debemos transformar ambos tipos de datos a un lenguaje común que la máquina pueda procesar. Aquí es donde entran los embeddings.

Los embeddings no son valores aleatorios, sino vectores generados por redes neuronales. En el caso del texto, cada palabra se convierte en un token que el modelo analiza en conjunto con las demás palabras para capturar el significado de la frase. Para las imágenes, se extraen patrones visuales mediante convoluciones o atención, que luego se combinan para representar toda la imagen como un vector único. Este proceso permite que tanto texto como imagen puedan compararse en un mismo espacio matemático

Un embedding es un vector que resume lo más importante de un texto o de una imagen. Para un texto, captura el significado de las palabras, cómo se relacionan entre sí y el contexto de la frase; para una imagen, refleja lo que hay en ella, como los objetos presentes, sus formas, colores, texturas o incluso relaciones espaciales entre elementos. Estos vectores permiten que la máquina trabaje con información compleja de forma matemática y comparativa, convirtiendo conceptos abstractos en coordenadas en un espacio multidimensional.

Para visualizarlo, se puede usar un ejemplo muy simple en dos dimensiones. Supóngase que se tiene la frase 'gato jugando' y una imagen de un gato jugando. Sus embeddings podrían ser:

$$t = [0.8\,, 0.6] \qquad i = [0.7\,, 0.7]$$

Donde t representa el vector del texto e i el vector de la imagen.

Antes de comparar, se aplicó normalización L2 a todos los embeddings. Esto significa que cada vector $v = (v_1, \dots, v_d)$ se divide por su norma euclídea $|v|_2 = \sqrt{v_1^2 + v_2^2 + \cdots + v_d^2}$ obteniendo el vector $\hat{v} = \frac{v}{|v|_2}$. Tras esta operación, todos los embeddings "viven" en la hiperesfera unitaria ($|\hat{v}|_2 = 1$), de modo que las comparaciones se basan únicamente en la dirección (contenido semántico) y no en el tamaño del vector. En la práctica, se evita normalizar vectores nulos; si existe riesgo numérico, se añade un ε muy pequeño al denominador.

Con los vectores ya normalizados, la similitud coseno entre dos representaciones A (texto) y B (imagen) se define como:

$$sim(A,B) = \frac{(A \cdot B)}{\left(\left|\left|A\right|\right|_2 \left|\left|B\right|\right|_2\right)}$$

donde $A \cdot B$ es el producto punto (suma de productos componente a componente) y $\left|\left|\cdot\right|\right|_2$ es la norma L2. Geométricamente, esta fracción equivale al coseno del ángulo entre A y B. Si ambos vectores están L2-normalizados ($\left|\left|A\right|\right|_2 = \left|\left|B\right|\right|_2 = 1$), la expresión se simplifica a:

$$sim(A,B) = A \cdot B$$

es decir, la similitud coseno coincide exactamente con el producto punto de los vectores unitarios.

El resultado teórico va de -1 a 1 (1 indica alineación máxima, 0 ausencia de relación y -1 oposición). En la mayoría de escenarios con embeddings normalizados (y a menudo no negativos) los valores prácticos tienden a concentrarse entre 0 y 1. Con esos puntajes se ordenaron las imágenes de mayor a menor similitud, obteniendo un ranking de las más cercanas a la descripción.[20]

En nuestro ejemplo, la similitud coseno se calcula así:

$$sim(i,t) = \frac{(0.8 \cdot 0.7 + 0.6 \cdot 0.7)}{\left(\left(\sqrt{(0.8^2 + 0.6^2)} \times \sqrt{(0.7^2 + 0.7^2)}\right)\right.} \approx 0.99$$

Un valor cercano a 1 indica que el texto y la imagen están muy alineados, es decir, que describen el mismo concepto de manera coherente. Por el contrario, si la imagen fuera completamente diferente, como un paisaje sin gatos, la similitud coseno sería mucho menor, mostrando que el texto y la imagen no coinciden.

2.2.2 Modelos de Lenguaje Visual (VLMs) y Modelos de Lenguaje de Gran Escala (LLMs)

En los últimos años, los avances en inteligencia artificial han estado marcados por el desarrollo de dos tipos de modelos fundamentales: los modelos de lenguaje de gran escala (LLMs, por sus siglas en inglés) y los modelos de lenguaje visual (VLMs). Estos modelos han abierto nuevas posibilidades en el tratamiento del lenguaje natural y su conexión con la información visual, siendo especialmente relevantes para sistemas que requieren comprender descripciones textuales y relacionarlas con imágenes, como es el caso del agente propuesto en este proyecto.

Los **LLMs**, como *BERT* o *GPT*, son redes neuronales entrenadas con enormes cantidades de texto para predecir la probabilidad de aparición de palabras o frases en función

del contexto. Su arquitectura basada en *Transformers* (arquitectura de redes neuronales que permite a los modelos de lenguaje comprender el significado contextual de las palabras) les permite comprender la estructura y significado del lenguaje natural con una profundidad que antes no era posible. Gracias a esto, estos modelos pueden interpretar descripciones humanas complejas, detectar atributos y transformar un lenguaje coloquial en una estructura más formal y operativa para un sistema automatizado.

Esta arquitectura *Transformer* supuso un cambio radical respecto a los modelos anteriores, como las redes recurrentes, que procesaban el texto palabra por palabra. Los *Transformers* permiten analizar todas las palabras de una frase al mismo tiempo, lo que acelera el procesamiento y mejora la comprensión del contexto. Además, incorporan un mecanismo de atención que les permite identificar qué partes del texto son más relevantes, incluso si están lejos entre sí. Esto les da una capacidad mucho mayor para entender relaciones complejas entre conceptos, generar respuestas coherentes y adaptarse al estilo del lenguaje humano con gran precisión.

En tiempos recientes, se han desarrollado modelos más sofisticados, entre ellos GPT-4 de OpenAI y Claude 3 de Anthropic, que destacan por su capacidad para interpretar con mayor precisión descripciones complejas y realizar razonamientos semánticos más afinados. Estas herramientas resultan especialmente útiles al convertir expresiones verbales cargadas de matices en formatos que pueden ser empleados en procesos de búsqueda o categorización. Además, su integración con sistemas que combinan texto e imagen es cada vez más frecuente, lo que potencia su aplicación en entornos multimodales.

Por otro lado, los **VLMs** como *CLIP* o *BLIP*, son modelos entrenados para vincular texto e imagen en un mismo espacio vectorial. Es decir, pueden analizar una imagen y una descripción escrita y determinar qué tan bien se corresponden entre sí. Esto se logra mediante un proceso de entrenamiento contrastivo, donde el modelo aprende a asociar correctamente pares de texto-imagen y a distinguirlos de combinaciones incorrectas. Esta capacidad resulta especialmente útil para tareas de búsqueda por descripción, ya que permite al sistema identificar qué rostros se ajustan mejor a una determinada descripción verbal.

La integración de ambos tipos de modelos en un mismo sistema representa una oportunidad única para el desarrollo de soluciones más flexibles e inteligentes. En el contexto de la seguridad, esto significa poder buscar a una persona únicamente con una descripción como “hombre joven con gorra azul y barba corta” y localizar en una base de imágenes aquella que mejor encaje con esa descripción, sin necesidad de contar con una fotografía previa.

Este tipo de tecnología no solo mejora las capacidades actuales de los sistemas de identificación, sino que también transforma el paradigma: ya no se trata solo de reconocer, sino de comprender el lenguaje humano y conectarlo con el entorno visual de forma contextual. Por este motivo, los VLMs y LLMs son pilares fundamentales en la arquitectura del agente de IA planteado en este trabajo.

1. **GPT-3** es uno de los modelos de lenguaje más grandes y potentes jamás entrenados. Demuestra que los modelos con muchos parámetros pueden aprender a realizar tareas complejas de lenguaje (traducción, preguntas,

resúmenes) con pocos ejemplos o incluso sin entrenamiento adicional (zero-shot/few-shot learning). Su gran capacidad de comprensión del lenguaje natural lo convierte en un referente para tareas que implican interpretar descripciones humanas. [1]

2. **GPT-4** es un modelo multimodal de última generación desarrollado por OpenAI. A diferencia de su predecesor, no solo procesa texto, sino que también puede interpretar imágenes, lo que lo hace especialmente útil para tareas que combinan lenguaje y visión. Destaca por su capacidad para comprender descripciones humanas complejas, generar respuestas coherentes y realizar tareas avanzadas de razonamiento. Su rendimiento en múltiples benchmarks supera ampliamente a modelos anteriores. [2]

3. **Claude 3** es una familia de modelos de lenguaje desarrollada por Anthropic, diseñada con un fuerte enfoque en alineación, comprensión contextual y razonamiento lógico. El modelo más avanzado, Claude 3 Opus, ha demostrado habilidades destacadas para interpretar descripciones verbales detalladas y convertirlas en inferencias precisas, lo que lo hace útil en sistemas que requieren interpretación de lenguaje natural aplicado a contextos visuales o de búsqueda semántica. [3]

4. **BERT** es un modelo de lenguaje que cambia la forma de preentrenar redes neuronales. A diferencia de modelos anteriores, BERT lee el texto en ambas direcciones (bidireccional), mejorando la comprensión contextual. Es muy eficaz para tareas como clasificación de texto, análisis de sentimientos, y respuestas a preguntas. Fue el punto de partida para muchos modelos LLM actuales. [4]

5. **CLIP** (Contrastive Language–Image Pretraining) es un modelo que puede entender imágenes y textos al mismo tiempo. Lo especial de CLIP es que no necesita entrenarse de nuevo para cada tarea, porque ya fue entrenado con cientos de millones de pares de texto e imagen que encontró en internet. Gracias a esto, puede hacer cosas muy útiles como buscar imágenes a partir de una descripción, reconocer rostros por características dadas en un texto o clasificar imágenes sin usar datasets específicos.

   CLIP funciona con dos redes que trabajan en paralelo: una para texto y otra para imágenes. La red de texto es un Transformer, un modelo que entiende el lenguaje analizando cómo cada palabra se relaciona con las demás. Primero, la frase se convierte en tokens, que son fragmentos más pequeños de texto (palabras o subpalabras) que el modelo puede manejar. Cada token se pasa por capas de atención, que son módulos que permiten que una palabra “mire” a todas las demás y asigne más peso a las más relevantes para su significado (por ejemplo, en la frase “gafas negras”, la palabra negras se relaciona más con gafas que con el resto). Tras varias de estas capas, el Transformer produce un vector que resume toda la frase. La red de imágenes puede ser una ResNet (que usa convoluciones para detectar patrones visuales como bordes, texturas o formas) o un Vision Transformer (ViT) (que divide la foto en parches, como cuadritos, y los procesa con

la misma atención que en texto, relacionando cada parte de la imagen con las demás). Después de varias transformaciones, también genera un vector que representa el contenido visual. Finalmente, ambos vectores se ajustan al mismo tamaño mediante una proyección lineal (una transformación matemática que reduce o expande la dimensión) y se normalizan. De esta forma, texto e imagen quedan en un espacio compartido, donde sus vectores pueden compararse directamente.

El secreto está en cómo se entrena. CLIP aprende comparando qué textos van con qué imágenes. Si una frase realmente describe a una imagen, sus vectores quedan más cercanos. Si no corresponden, se alejan. El modelo se entrena justamente para que las parejas correctas tengan siempre la mayor similitud. Para lograrlo, usa una función matemática llamada pérdida contrastiva, que premia cuando los pares correctos están cerca y castiga cuando se confunde.

Una vez entrenado, CLIP se puede usar de inmediato en tareas prácticas. Supongamos que tenemos una base de fotos de rostros y quieres encontrar a alguien a partir de una descripción como “persona con gafas negras y barba”. CLIP convierte esa frase en un vector y también convierte todas las fotos en vectores. Luego compara la frase con cada foto midiendo la similitud entre sus vectores. La foto cuyo vector esté más cerca del vector de la frase es la que mejor coincide con la descripción. [5]

6. **BLIP** (Bootstrapped Language-Image Pretraining) es un modelo de visión-lenguaje diseñado para entender y generar texto a partir de imágenes mediante un entrenamiento en dos fases complementarias. La primera fase se enfoca en alinear imágenes con sus textos correspondientes. Para ello, BLIP convierte cada imagen en un vector que representa su contenido visual y cada texto en un vector que resume su significado semántico. Luego, mediante un aprendizaje contrastivo, acerca los vectores de las imágenes con sus textos correctos y aleja los vectores de los pares incorrectos. Matemáticamente, si i es el vector de la imagen y t el vector del texto, BLIP maximiza la similitud coseno $i \cdot t$ para pares correctos y la minimiza para pares incorrectos. Esta fase permite que el modelo capture relaciones básicas entre la información visual y lingüística de manera robusta.

   La segunda fase de BLIP se centra en generación de lenguaje a partir de imágenes, conocida como image captioning, y en tareas de completado de texto visual. Aquí, el modelo utiliza los vectores aprendidos en la primera fase como base y entrena un decodificador de lenguaje (por ejemplo, un Transformer) que toma como entrada la representación visual de la imagen y produce texto coherente. Esto implica que BLIP no solo reconoce si un texto corresponde a una imagen, sino que también puede inferir detalles visuales, relacionarlos semánticamente y expresarlos en palabras. Por ejemplo, ante una foto de un perro jugando en el parque, BLIP puede generar “Un perro corre feliz entre la hierba”, mostrando comprensión contextual y gramatical.

En cuanto a la estructura del modelo, BLIP combina componentes de visión y lenguaje de manera integrada. La red de imágenes puede ser una ResNet o un Vision Transformer, que transforma la imagen en vectores de características; mientras que la parte de texto es un Transformer que procesa tokens y genera embeddings semánticos. Durante la primera fase, estos embeddings se alinean mediante proyecciones y normalizaciones para que estén en un mismo espacio de representación compartido, facilitando la comparación y asociación de pares imagen-texto. En la segunda fase, se usa un decodificador autoregresivo que genera palabras secuencialmente basándose en la información visual, aplicando atención para decidir qué partes de la imagen son relevantes en cada paso de la generación de texto.

Gracias a esta estrategia en dos fases, BLIP supera a modelos que solo usan aprendizaje contrastivo, porque combina la capacidad de alinear y comparar vectores con la habilidad de generar y completar texto de manera coherente. Esto le permite realizar tareas como búsqueda de imágenes por texto, captioning automático y razonamiento visual-textual, captando relaciones complejas entre la información visual y lingüística que modelos anteriores no podían inferir directamente.[6]

7. **LLaVA (Large Language and Vision Assistant)** combina modelos de lenguaje grandes (LLMs) con capacidades de percepción visual al integrar un codificador de imágenes (como CLIP o EVA) y un modelo tipo LLaMA. Su funcionamiento se basa en el ajuste fino con instrucciones visuales (Visual Instruction Tuning), lo que le permite responder preguntas sobre imágenes, generar descripciones, y entender interacciones complejas entre objetos y escenas. Es especialmente útil para tareas como captioning avanzado, VQA (Visual Question Answering), o búsqueda basada en lenguaje natural. LLaVA representa un paso hacia asistentes multimodales más completos. [7]

2.3 Privacidad y consideraciones éticas en el uso de IA para la identificación facial

El desarrollo de sistemas de inteligencia artificial aplicados a la identificación facial, especialmente aquellos que combinan lenguaje e imagen, plantea importantes desafíos éticos y de privacidad. En un contexto tan sensible como el de la seguridad, donde el tratamiento de datos personales puede afectar directamente los derechos y libertades de las personas, resulta imprescindible considerar las implicaciones del uso de estas tecnologías.

Uno de los principales riesgos asociados al uso de IA en vigilancia es el potencial abuso de la vigilancia masiva. La posibilidad de identificar personas en tiempo real, incluso a partir de descripciones, puede derivar en situaciones donde se vulneren derechos fundamentales como la libertad de movimiento, la privacidad o el anonimato. Además, el uso de sistemas automatizados sin supervisión humana puede llevar a decisiones erróneas o discriminatorias si

los algoritmos presentan sesgos en los datos de entrenamiento, como ha ocurrido en múltiples estudios sobre reconocimiento facial que evidencian tasas de error más elevadas en personas racializadas o en mujeres.

Desde el punto de vista legal, existen regulaciones específicas que buscan garantizar un uso responsable de estas tecnologías. En el ámbito europeo, el Reglamento General de Protección de Datos (RGPD) establece que el tratamiento de datos biométricos, como los rasgos faciales, solo puede realizarse bajo condiciones estrictas y con un consentimiento explícito o una base legal sólida [8]. Además, la propuesta de Ley de Inteligencia Artificial de la Unión Europea introduce restricciones al uso de sistemas de identificación biométrica remota en espacios públicos, especialmente si son en tiempo real.[9]

Asimismo, se plantea la importancia de incorporar mecanismos de explicabilidad y transparencia, de modo que se pueda comprender cómo el sistema ha llegado a una determinada sugerencia o resultado. Esto es clave para fomentar la confianza en el uso de IA en ámbitos críticos, y forma parte de las recomendaciones actuales tanto desde el punto de vista técnico como regulatorio.[10]

En conclusión, aunque el uso de agentes de IA en tareas de identificación facial ofrece grandes ventajas operativas, su desarrollo debe ir siempre acompañado de una reflexión ética rigurosa y de medidas concretas que garanticen un uso justo, seguro y respetuoso con los derechos humanos.

2.4 Panorama de los sistemas actuales y propuesta alternativa.

Hoy, la búsqueda e identificación visual que realizan administraciones y cuerpos policiales se apoya en varios tipos de sistemas. El más común son las grandes bases biométricas: repositorios con millones de fotografías oficiales (normalmente fichas policiales) donde, ante una imagen o una descripción, se pide una lista de candidatos ordenada por parecido. Ejemplos conocidos son el programa de identificación por foto del FBI (NGI-IPS) o el sistema de reconocimiento facial de INTERPOL (IFRS), al que contribuyen distintos países con sus propios registros. Estos sistemas emplean técnicas de IA y aprendizaje automático de visión por computador: redes neuronales extraen una “huella” o vector de cada rostro y lo comparan en bases vectoriales para devolver coincidencias ordenadas por similitud, aplicando umbrales de calidad y, por lo general, con revisión humana antes de cualquier decisión. Estos sistemas funcionan “en diferido”: no están mirando cámaras en tiempo real, sino que responden a consultas concretas sobre una base ya almacenada.

En paralelo, algunos cuerpos han probado o utilizan reconocimiento “en vivo” en ubicaciones concretas: cámaras que, durante un operativo, comparan los rostros de las personas que pasan con listas cerradas (por ejemplo, personas buscadas para un caso). Es una tecnología pensada para escenarios muy específicos y, por su potencial impacto en derechos y posibles sesgos, está sujeta a controles estrictos. A su lado conviven herramientas no biométricas pero muy extendidas, como los lectores automáticos de matrículas (ALPR), que permiten detectar y seguir vehículos a partir de su placa, generando alertas cuando hay coincidencias con listas de interés. Y, en un plano distinto, existen servicios comerciales

basados en raspado masivo de imágenes de Internet (recogen fotos sin pedir permiso para alimentar buscadores faciales), que han recibido sanciones y críticas por falta de base legal y por sus riesgos para la privacidad.

Este contexto ha llevado a que la Unión Europea marque límites claros: el AI Act restringe de forma muy severa la identificación biométrica en tiempo real en espacios públicos, reservándola a supuestos muy concretos y con salvaguardas reforzadas (autorizaciones previas, supervisión, trazabilidad). La tendencia regulatoria, en general, va hacia pedir proporcionalidad, transparencia y controles humanos sobre estas tecnologías.

Frente a ese enfoque amplio y, a menudo, complejo, la solución propuesta adopta un camino acotado y proporcional. En lugar de vigilar de forma continua o rastrear la red, trabaja bajo demanda sobre un repositorio propio y documentado de imágenes: se busca cuando hace falta, sobre datos cuyo origen y finalidad son conocidos. Este diseño facilita la transparencia (queda registro de quién buscó, con qué criterios y qué resultados obtuvo), incorpora privacidad por diseño (finalidades y plazos de conservación definidos) y reduce los falsos positivos al limitar el universo de búsqueda y exigir revisión humana antes de cualquier decisión.[24]

Capítulo 3. OBJETIVOS

Este capítulo expone los propósitos del proyecto y los hitos que guiarán su desarrollo. Definir claramente los objetivos resulta esencial porque marcan el rumbo de todo el trabajo: ayudan a priorizar esfuerzos, orientan la toma de decisiones técnicas y sirven como referencia para evaluar el éxito final.

Más allá de lo técnico, los objetivos también reflejan la motivación principal del proyecto: demostrar que es posible acercar los últimos avances en inteligencia artificial multimodal a un caso de uso concreto y útil, como es la búsqueda de rostros a partir de descripciones verbales.

3.1 Objetivo general

El objetivo central de este trabajo es desarrollar un sistema capaz de interpretar descripciones en lenguaje natural y usarlas para buscar rostros humanos en una base de datos de imágenes**,** con el fin de devolver los perfiles más similares.

Se trata de cubrir una carencia concreta: en la actualidad, los sistemas de reconocimiento facial suelen basarse en imágenes de referencia, lo que limita su uso en situaciones donde no hay fotografías disponibles. Este proyecto busca dar un paso más allá y explorar cómo las palabras pueden convertirse en una herramienta para acceder a información visual, aprovechando modelos avanzados de procesamiento de lenguaje e imagen como BLIP y CLIP, y combinándolos dentro de una solución funcional.

Este objetivo, aunque técnico, tiene un fuerte componente social y práctico. Poder buscar rostros a partir de descripciones verbales abre oportunidades en ámbitos muy distintos: desde investigaciones policiales o análisis forense, hasta entornos de accesibilidad o aplicaciones comerciales donde los usuarios describen lo que buscan en lugar de subir imágenes.

3.2 Objetivos específicos

Para lograr este objetivo general, se han planteado una serie de metas más concretas que organizan el trabajo en fases y facilitan su desarrollo:

- **Analizar el estado del arte en modelos de lenguaje e imagen**, revisando opciones como CLIP, BLIP-2 o GPT-4, con el fin de determinar cuáles ofrecen mejores resultados al relacionar descripciones verbales con rostros humanos.

- **Construir una base de datos de imágenes de referencia**, adaptada a los fines del proyecto y respetando criterios éticos, legales y de privacidad. Esta base será la "prueba de fuego" sobre la que el sistema mostrará su capacidad para recuperar rostros.
- **Diseñar un espacio de representación común entre texto e imagen**, es decir, encontrar una manera de que ambos tipos de datos puedan compararse en igualdad de condiciones, mediante embeddings y métricas de similitud.
- **Integrar salvaguardas de privacidad y seguridad desde el diseño**, alineadas con regulaciones como el AI Act y el RGPD, asegurando que la tecnología se utilice de manera responsable y con trazabilidad.
- **Desarrollar un backend robusto**, encargado de la lógica de comparación y recuperación de imágenes, y un frontend sencillo que permita a cualquier usuario introducir descripciones y visualizar resultados sin necesidad de conocimientos técnicos.
- **Desplegar la solución en la nube**, utilizando servicios de Azure que permitan accesibilidad global, escalabilidad y estabilidad, además de un proceso de actualización continuo mediante integración con GitHub.
- **Realizar pruebas experimentales exhaustivas**, evaluando la precisión del sistema con diferentes descripciones, midiendo métricas de similitud y analizando tanto los aciertos como las limitaciones.
- **Reflexionar críticamente sobre los resultados**, registrando no solo lo que funcionó, sino también las limitaciones encontradas, y proponiendo mejoras o líneas futuras que den continuidad al proyecto.

3.3 Beneficios del proyecto

Este trabajo no busca únicamente demostrar un concepto técnico, sino también aportar beneficios tangibles en varios niveles:

1. **Utilidad práctica**: ofrece una forma de realizar búsquedas faciales a partir de texto, algo especialmente valioso en escenarios donde no existe una imagen de partida, como declaraciones de testigos o descripciones verbales en procesos de investigación.
2. **Accesibilidad**: al estar disponible en forma de aplicación web, el sistema puede ser usado por cualquier persona con un navegador, sin necesidad de conocimientos avanzados ni instalaciones complejas.
3. **Innovación tecnológica**: combina de manera novedosa modelos de lenguaje e imagen dentro de un flujo de trabajo completo, desde la descripción hasta la recuperación de rostros, mostrando cómo la inteligencia artificial multimodal puede resolver problemas reales.
4. **Escalabilidad**: gracias al despliegue en la nube, la solución puede crecer fácilmente, tanto en número de imágenes como en capacidad de procesamiento, sin necesidad de grandes cambios en la arquitectura.
5. **Base para futuros proyectos**: el prototipo abre la puerta a ampliaciones como incluir audio en lugar de texto, aceptar descripciones más largas y detalladas, mejorar la

calidad de los embeddings con modelos más recientes, o extender la búsqueda a otros dominios visuales (objetos, escenas, estilos).

Capítulo 4. DESARROLLO DEL PROYECTO

4.1 Planificación del proyecto

Este proyecto se desarrolló en varias etapas, con el objetivo de construir un sistema capaz de recuperar imágenes a partir de descripciones generadas automáticamente. La idea principal era comprobar si, partiendo de una descripción textual obtenida de una imagen, se podía identificar esa misma imagen usando técnicas de búsqueda semántica.

En un inicio, se definieron claramente los objetivos y se eligió un dataset de imágenes faciales. La elección de este dataset se basó en su diversidad: diferentes expresiones, poses y rasgos faciales que permitían evaluar de manera efectiva si el sistema podía generar descripciones precisas y reconocer correctamente cada imagen.

Luego se diseñó la arquitectura del sistema, basada en tres componentes principales. Primero, BLIP se encargó de generar automáticamente descripciones de cada imagen, capturando sus detalles más relevantes. Después, CLIP transformó tanto las imágenes como los textos en vectores dentro de un espacio común, permitiendo medir qué tan cercanas eran semánticamente. Finalmente, una función de similitud coseno comparó estos vectores para identificar cuál imagen coincidía más con la descripción generada.

El sistema se implementó de manera distribuida. El frontend se desarrolló y alojó en Azure Web Service, con el código manejado desde GitHub, ofreciendo una interfaz sencilla y accesible desde cualquier navegador. Por su parte, el backend se ejecutó en una máquina virtual de Azure, donde se corrieron los modelos y se realizaron los cálculos de similitud. Esta separación permitió que cada parte del sistema pudiera actualizarse o escalarse de forma independiente, haciendo el proceso más eficiente.

Finalmente, se analizaron los resultados y se identificaron tanto los aciertos como las áreas de mejora. El proyecto demostró cómo la combinación de modelos de visión y lenguaje puede facilitar la búsqueda de imágenes a partir de texto, abriendo posibilidades interesantes para aplicaciones como búsqueda de imágenes, etiquetado automático o sistemas de recomendación basados en contenido visual.

4.2 Base de datos

Las imágenes empleadas en el sistema provienen del conjunto de datos Synthetic Faces High Quality (SFHQ), disponible públicamente en Kaggle. Se trata de un repositorio de más de 425.000 retratos sintéticos en alta resolución (1024×1024 píxeles), generados mediante modelos de inteligencia artificial. Al no corresponder a personas reales, su uso no presenta implicaciones éticas o legales relacionadas con datos personales, lo que facilita el cumplimiento normativo sin sacrificar calidad visual.

Para este proyecto se seleccionó aleatoriamente una muestra de 550 imágenes del SFHQ. La elección aleatoria se realizó con el objetivo de reducir sesgos y obtener un subconjunto representativo de la variedad del dataset (rasgos faciales, edades aparentes, peinados, iluminación y fondos). Estas 550 imágenes se utilizaron como base visual sobre la que se efectuaron búsquedas a partir de descripciones escritas (por ejemplo, “mujer joven con gafas redondas y pelo rizado, mirando a cámara”), evaluando así la capacidad del sistema para traducir texto en consultas que recuperen los rostros sintéticos más cercanos.[11]

A continuación se presentan ejemplos representativos de las 550 imágenes seleccionadas del SFHQ. Los casos se han elegido para ilustrar la diversidad del conjunto (edad aparente, rasgos faciales, accesorios, pose, iluminación y fondo):

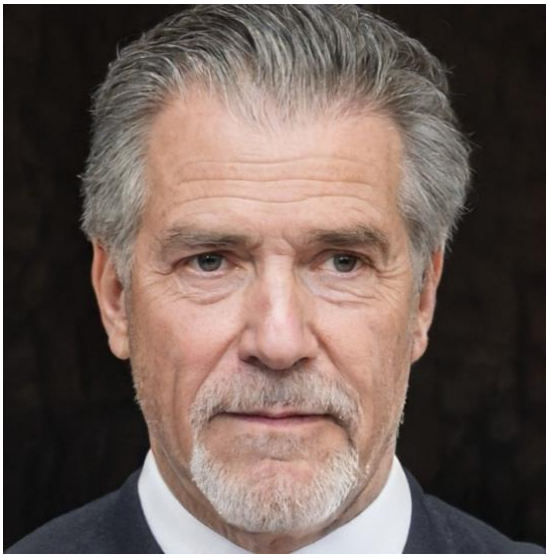

Figura 1. Imagen 150.

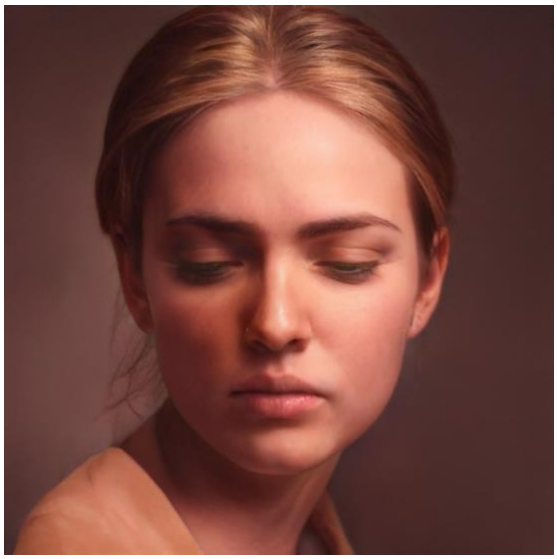

Figura 2. Imagen 347.

Figura 1. Imagen 459.

Figura 4. Imagen 495.

4.3 Descripción de la solución, metodologías y herramientas empleadas

La idea principal de este proyecto es comprobar si una descripción generada automáticamente a partir de una imagen es lo suficientemente precisa como para que, al usarla como consulta, el sistema devuelva esa misma imagen como la más similar. Para ello, se combinan modelos de visión por computador y lenguaje natural, junto con herramientas de comparación semántica basadas en embeddings.

El proceso comienza con un conjunto de imágenes faciales. Cada imagen se analiza mediante un modelo generador de descripciones (en este caso, BLIP), que produce una frase en lenguaje natural representando el contenido visual. A continuación, todas las descripciones se convierten en vectores utilizando un modelo de embeddings semánticos. Los embeddings semánticos son representaciones vectoriales de textos que capturan su significado contextual, estos son especialmente útiles para tareas como búsqueda de información, clasificación de textos, detección de duplicados o comparación semántica, ya que facilita medir el grado de similitud entre textos, incluso cuando no comparten palabras exactas.

En paralelo, también es posible convertir las imágenes en vectores visuales mediante modelos como CLIP, lo que permitiría comparar directamente imágenes y descripciones en un mismo espacio latente.

Con todos los vectores generados, se calcula la similitud coseno entre la descripción original y el resto del dataset.

El desarrollo se ha realizado íntegramente en Python, utilizando modelos disponibles a través de la plataforma Hugging Face, y librerías como transformers, sentence-transformers, NumPy, pandas y scikit-learn. La arquitectura se ha diseñado de forma modular, permitiendo validar paso a paso cada componente del sistema (generación, vectorización y búsqueda).

4.3.1 Arquitectura general de la solución

La solución propuesta parte de una idea central: utilizar descripciones generadas automáticamente como medio para recuperar imágenes en un conjunto amplio, basándose en la similitud semántica entre texto e imagen. Para ello, se ha diseñado una arquitectura que combina modelos de lenguaje natural, visión por computador y técnicas de comparación vectorial en un espacio multimodal compartido.

El proceso comienza con un conjunto de imágenes faciales. Cada una de ellas es procesada mediante un modelo de generación de lenguaje natural con capacidades visuales (en este caso, BLIP), que produce una descripción textual que resume sus rasgos más representativos. Estas descripciones, generadas previamente y almacenadas, sirven como punto de partida para la búsqueda.

Una vez disponibles las descripciones, se transforman en vectores semánticos utilizando el modelo CLIP en su modalidad textual. En paralelo, todas las imágenes del conjunto han sido convertidas en vectores mediante CLIP en su modalidad visual. Esto permite

que tanto imágenes como descripciones estén representadas dentro de un mismo espacio vectorial multimodal, diseñado para facilitar comparaciones entre lenguaje e imagen.

La elección de CLIP y BLIP como modelos principales en este trabajo responde a una combinación de eficacia, accesibilidad y adecuación a los objetivos planteados. CLIP, desarrollado por OpenAI, destaca por su capacidad para representar texto e imagen en un mismo espacio vectorial, lo que facilita la comparación directa entre descripciones verbales y rostros. Por su parte, BLIP aporta una funcionalidad clave: la generación automática de descripciones a partir de imágenes, permitiendo enriquecer la base de datos y habilitar búsquedas más precisas. Aunque existen otros modelos más recientes y avanzados, muchos de ellos requieren recursos computacionales más exigentes o no cuentan aún con una documentación y soporte comunitario tan sólido. En cambio, CLIP y BLIP son modelos abiertos, ampliamente validados y con entornos de desarrollo bien establecidos, lo que los convierte en una elección práctica y fiable para el desarrollo de un sistema de búsqueda facial basado en descripciones textuales.

En la fase final del proceso, se calcula la similitud coseno entre el vector de cada descripción y los vectores de todas las imágenes del conjunto. El sistema devuelve un ranking de imágenes ordenadas según su grado de similitud con la descripción introducida, priorizando aquellas que mejor se alinean semánticamente con el contenido textual.

Es importante destacar que, dado que la generación de descripciones y la recuperación por similitud se realizan mediante modelos distintos (BLIP y CLIP, respectivamente), no se garantiza que la imagen original que dio lugar a la descripción ocupe necesariamente los primeros puestos del ranking. Sin embargo, el sistema permite evaluar la eficacia de las descripciones como herramienta de búsqueda y ofrece una aproximación funcional al problema de recuperación de imágenes basada en lenguaje natural.

En resumen, el sistema transforma imágenes en texto y utiliza esas descripciones como consultas para recuperar imágenes similares. Este enfoque permite evaluar, de manera indirecta, la calidad semántica de las descripciones generadas automáticamente y su utilidad en tareas de búsqueda y recuperación visual.

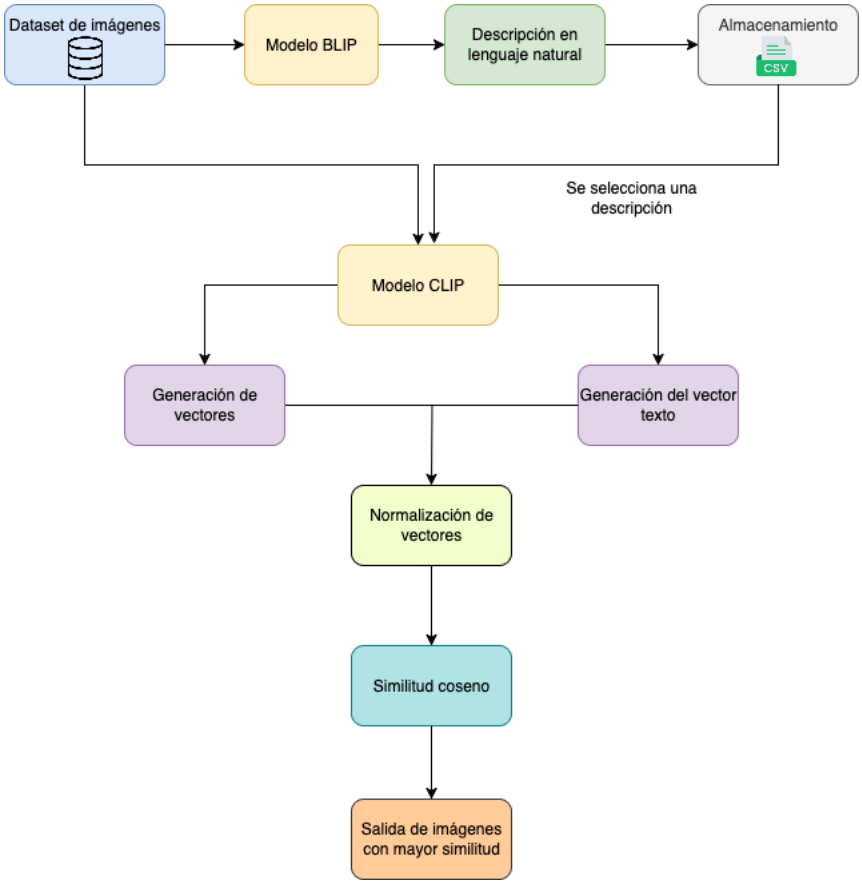

Figura 5. Arquitectura de la solución.

4.3.2 Modelos y tecnologías utilizados

Para el desarrollo de este sistema, se combinaron distintos modelos de inteligencia artificial especializados en lenguaje natural y visión por computador, junto con herramientas de programación que permitieran automatizar todo el flujo de trabajo.[12] El objetivo principal fue diseñar un sistema capaz de generar descripciones automáticas a partir de imágenes y, posteriormente, utilizar dichas descripciones como entrada para recuperar imágenes semánticamente similares dentro de un conjunto amplio.[13]

El primer componente del sistema fue un modelo de generación de descripciones basado en visión-lenguaje, concretamente BLIP (Bootstrapped Language-Image Pretraining).[15] Este modelo, desarrollado por Salesforce, es capaz de interpretar imágenes y generar de forma automática descripciones en lenguaje natural. A través de un script en Python, cada imagen del conjunto fue procesada por BLIP, que generó una frase descriptiva asociada. Estas descripciones se almacenaron en un archivo CSV, lo que permitió desacoplar la fase de generación de la fase de búsqueda posterior.[16]

Una vez generadas y almacenadas las descripciones, se utilizó el modelo CLIP (Contrastive Language-Image Pretraining), desarrollado por OpenAI, para representar tanto las

descripciones como las imágenes dentro de un espacio vectorial multimodal compartido. Este modelo permite comparar texto e imagen directamente mediante la similitud coseno entre sus embeddings. En esta etapa, las descripciones fueron transformadas en vectores utilizando la modalidad textual de CLIP, mientras que las imágenes fueron procesadas por la modalidad visual del mismo modelo.[17]

Es importante destacar que, aunque la descripción de entrada haya sido generada a partir de una imagen concreta, el modelo CLIP no tiene conocimiento explícito de este vínculo. Por tanto, no se garantiza que la imagen original ocupe las primeras posiciones del ranking de similitud. No obstante, si el sistema funciona correctamente, las imágenes más similares semánticamente a la descripción deberían compartir rasgos visuales clave, reflejando así la coherencia entre lenguaje e imagen.[18]

La comparación entre la descripción en texto y cada imagen se realizó mediante similitud coseno aplicada a sus vectores de representación (embeddings). De forma intuitiva, esta medida observa el ángulo entre dos vectores: cuanto más pequeño es ese ángulo, más alineadas están sus direcciones y, por tanto, más “parecidas” resultan a nivel semántico. Una ventaja clave es que la similitud coseno no depende de la magnitud de los vectores, solo de su dirección, lo que evita que diferencias de escala distorsionen la comparación.[19]

4.4 Implementación

El sistema fue desarrollado íntegramente en Python, utilizando modelos de inteligencia artificial distribuidos a través de la plataforma Hugging Face. El flujo de trabajo parte de un conjunto de imágenes almacenadas localmente, que se procesan de forma automática mediante dos componentes principales: generación de descripciones y recuperación por similitud.

En la primera fase, cada imagen es analizada mediante el modelo BLIP (*Bootstrapped Language-Image Pretraining*), especializado en tareas de visión-lenguaje. Este modelo genera una descripción textual breve que resume el contenido visual de la imagen, la cual se almacena en un archivo CSV para su uso posterior. [21]

En la segunda etapa, se emplea el modelo CLIP (*Contrastive Language-Image Pretraining*) para transformar tanto las descripciones como las imágenes en vectores dentro de un espacio semántico común. Esto permite evaluar el grado de similitud entre una descripción dada y todas las imágenes del conjunto utilizando la similitud coseno como métrica de comparación. [22]

En este código, los embeddings tienen 512 dimensiones, ya que el *modelo openai/clip-vit-base-patch32* de CLIP representa tanto texto como imágenes en vectores de ese tamaño. El embedding del texto (la descripción base) se almacena en la variable embedding_texto, que es un tensor de forma [1, 512]. Los embeddings de cada imagen se calculan individualmente y se van guardando en la lista embeddings_imagenes; después, todos esos vectores se concatenan en un único tensor llamado embeddings_tensor, cuya forma final es [n_imagenes_validas, 512], donde n_imagenes_validas es la cantidad de imágenes procesadas exitosamente. Así,

cada fila de embeddings_tensor corresponde a una imagen, y el embedding de texto queda en embedding_texto, ambos listos para compararse con la similitud de coseno.

A partir de la descripción seleccionada como consulta, se genera su correspondiente vector textual y se calcula su cercanía con los vectores visuales previamente obtenidos. El sistema devuelve un ranking de imágenes ordenadas según su alineación semántica con el texto.

Dado que los modelos utilizados no comparten parámetros ni están ajustados conjuntamente, no se espera que la imagen original esté necesariamente en la primera posición del ranking. Sin embargo, el hecho de que imágenes visualmente coherentes con la descripción aparezcan en los primeros puestos permite evaluar la efectividad del sistema como herramienta de recuperación basada en lenguaje natural.

Tras crear las descripciones y tras transformarse en vectores semánticos, el sistema compara cada descripción con el resto utilizando la métrica de similitud coseno. En este caso, se tomó la primera descripción generada automáticamente por el modelo (a woman with blonde hair and blue eyes) como punto de partida para buscar las descripciones más similares en todo el conjunto.

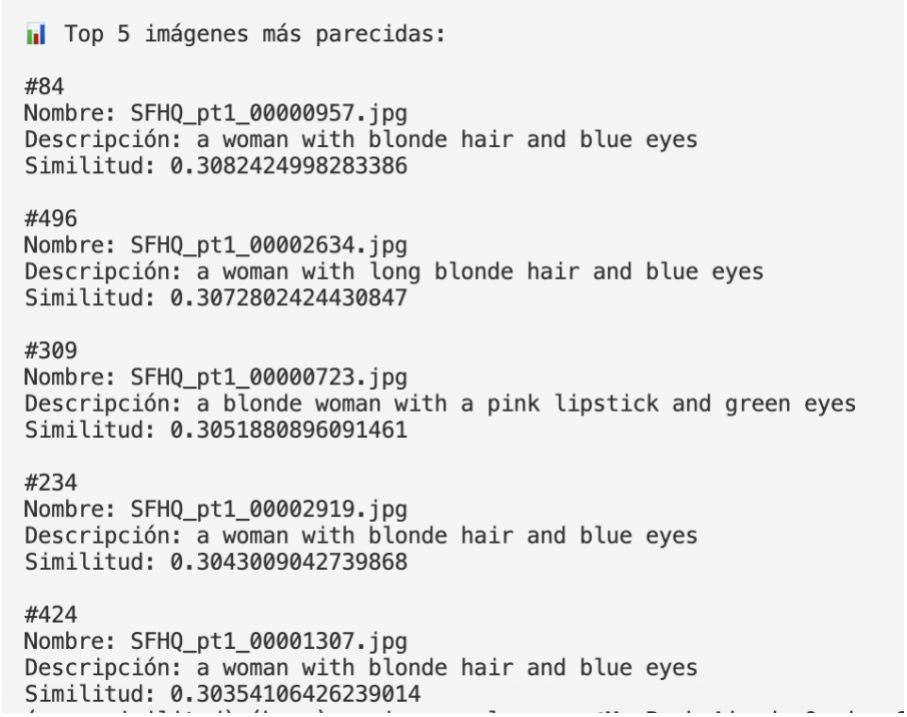

Figura 6. Resultados

Como se observa en los resultados, las descripciones generadas para las imágenes más similares son muy parecidas, aunque no completamente idénticas. Las puntuaciones de similitud coseno se sitúan alrededor de 0.30, lo que refleja una moderada cercanía semántica entre los textos comparados. Este valor indica que, si bien el modelo ha identificado ciertas

coincidencias en el contenido visual y textual (por ejemplo, el color de ojos y cabello), existen pequeñas variaciones léxicas o estructurales en las descripciones que afectan a la puntuación final.

El sistema utiliza como núcleo un modelo de generación de descripciones de imágenes (BLIP), que transforma contenido visual en texto. Luego, para la recuperación semántica, se emplea un modelo basado en embeddings (como Sentence Transformers), que representa cada descripción como un vector en un espacio de alta dimensión. La similitud coseno entre estos vectores permite medir cuán similares son dos descripciones, independientemente de su longitud o estructura gramatical.

El hecho de que la imagen original aparezca en la primera posición (#84) confirma que el sistema está funcionando correctamente en cuanto a consistencia entre visión y lenguaje: el modelo BLIP genera una descripción que se traduce en una representación semántica que coincide con la de la consulta.

No obstante, este resultado también evidencia una limitación importante del sistema: la tendencia del modelo generador a producir descripciones genéricas o poco específicas (por ejemplo, “a woman with blonde hair and blue eyes”) puede provocar que múltiples imágenes distintas reciban textos casi idénticos. Esto reduce el margen de discriminación semántica, especialmente en conjuntos de datos donde las diferencias entre imágenes son sutiles (como rostros con características similares).

En escenarios donde se requiera una mayor precisión (por ejemplo, identificación de personas en contextos reales), este comportamiento puede afectar negativamente el rendimiento del sistema. Para mejorar este aspecto, sería necesario:

- Incorporar modelos generativos más detallados o adaptados al dominio específico. Aplicar mecanismos de enriquecimiento de descripciones.

- O bien complementar la búsqueda semántica textual con características visuales adicionales directamente extraídas del embedding visual.

A continuación, se muestran las cinco imágenes cuya descripción generada es más similar a la de la imagen seleccionada, según la métrica de similitud coseno. Estas representan los resultados que el sistema considera más cercanos semánticamente a la consulta.

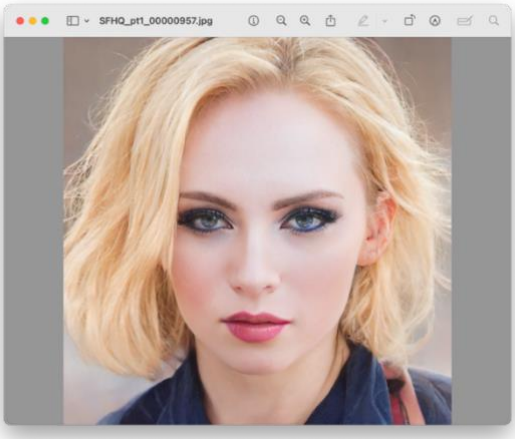

Figura 7. Imagen 957.

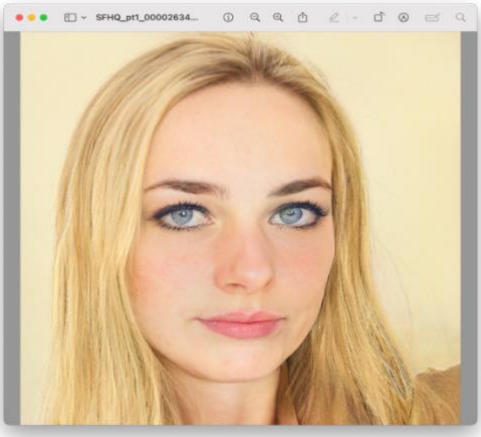

Figura 8. Imagen 2634.

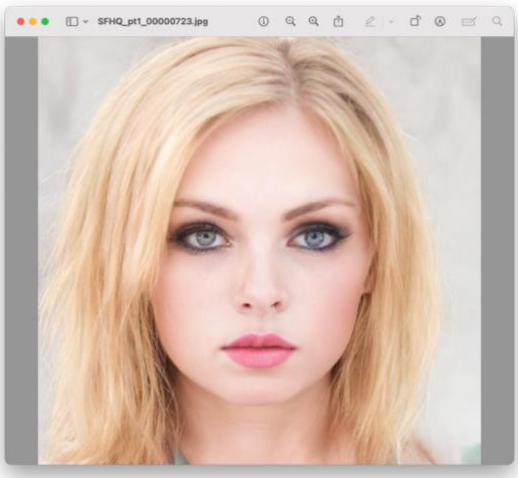

Figura 9. Imagen 723.

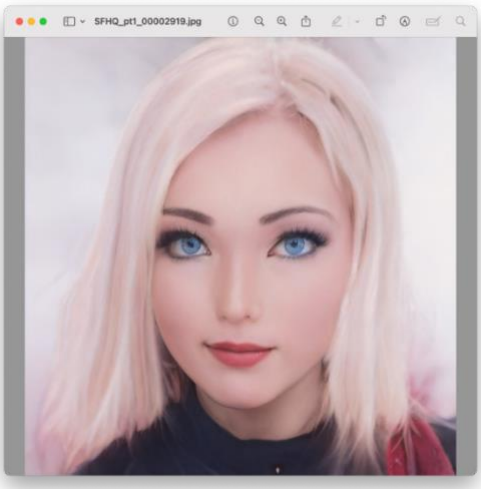

Figura 10. Imagen 2919.

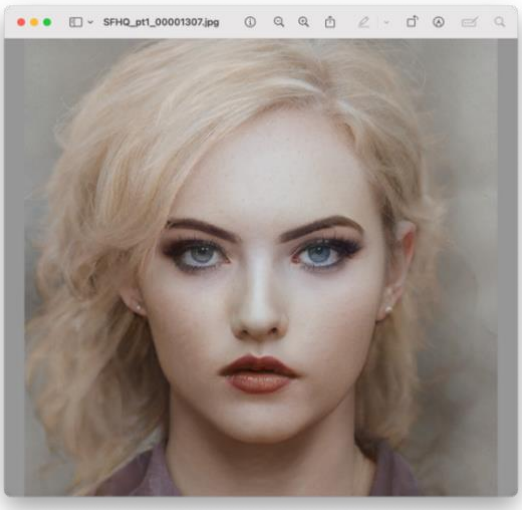

Figura 11. Imagen 1307.

Si este sistema se aplicara a un conjunto de millones de imágenes, el reto principal sería la eficiencia en el procesamiento y la búsqueda. Aunque el uso de vectores y la métrica de similitud coseno permite comparar descripciones de forma rápida, el número de comparaciones crecería exponencialmente con el tamaño del dataset. Para hacer frente a este problema, sería necesario implementar soluciones de escalabilidad como bases de datos vectoriales (por ejemplo, FAISS o ChromaDB), que permiten realizar búsquedas aproximadas mucho más rápidas. Además, también sería recomendable optimizar el preprocesamiento, almacenar los vectores ya generados y, si es posible, realizar las búsquedas en paralelo o mediante sistemas distribuidos. Estas mejoras permitirían mantener tiempos de respuesta aceptables incluso con volúmenes masivos de datos. [23]

4.5 Despliegue de la solución a través de una aplicación web

Esta aplicación ha sido creada para facilitar la búsqueda de personas a partir de una descripción escrita, combinando una interfaz web moderna y un sistema inteligente de reconocimiento de imágenes. El desarrollo se realizó con apoyo de GitHub Copilot, que asistió en la generación de código, sugerencias y estructuras base durante todo el proceso.

En el frontend, se agrupan los archivos que gestionan la experiencia visual y la interacción del usuario. El archivo principal index.html contiene la estructura de la página, mientras que styles.css define el diseño, los colores y la disposición de los elementos para que la app sea atractiva y fácil de usar. Los archivos JavaScript, como index.js e index.service.js, se encargan de la lógica de la aplicación: reciben la descripción que escribe el usuario, muestran la barra de carga animada y se comunican con el backend para obtener los resultados. fragmentos reutilizables.

Por otro lado, el backend es el motor que realiza la búsqueda inteligente. Aquí destacan archivos como api.py, que recibe las peticiones del frontend y responde con los resultados, y busqueda_imagnes.py, donde se encuentra la lógica que compara la descripción con las imágenes disponibles usando modelos de inteligencia artificial. El backend también cuenta con archivos de configuración como requirements.txt, que lista todas las dependencias necesarias para que el sistema funcione correctamente, y una carpeta de imágenes que sirve como base de datos visual.

En conjunto, la aplicación une una interfaz sencilla y moderna con un sistema de búsqueda avanzado, permitiendo que cualquier usuario pueda encontrar imágenes que se ajusten a una descripción, todo de manera rápida y visual. El uso de GitHub Copilot contribuyó a un desarrollo más ágil y consistente, manteniendo la claridad del código y reduciendo tiempos de implementación.

Para garantizar que la aplicación sea accesible, segura y fácil de mantener, se decidió desplegarla en Azure, aprovechando la amplia gama de servicios gestionados que ofrece la plataforma. El frontend se alojó en Azure Static Web Apps, lo que permite que los archivos se publiquen automáticamente cada vez que se actualiza el repositorio en GitHub. Esta integración asegura que cualquier cambio en el código se refleje de manera inmediata en la web, sin necesidad de intervenciones manuales ni procesos de compilación adicionales. La

plataforma también gestiona automáticamente el tráfico HTTPS, ofreciendo conexiones seguras y evitando problemas de certificados, lo que facilita que los usuarios accedan a la aplicación desde cualquier red de forma confiable.

El backend se ejecuta en una máquina virtual de Azure, donde Nginx actúa como servidor frontal recibiendo las peticiones y redirigiéndolas a Gunicorn, que ejecuta la aplicación Python encargada de procesar las solicitudes. La máquina virtual se encuentra protegida mediante un grupo de seguridad de red y un firewall interno, configurados para permitir únicamente el tráfico necesario y bloquear accesos no autorizados. Además, cuenta con una dirección IP pública que permite que las solicitudes lleguen desde Internet y se puedan vincular a un nombre de dominio para garantizar HTTPS válido. Este enfoque separa claramente la interfaz de usuario del motor de procesamiento, manteniendo la seguridad y la estabilidad del sistema mientras se asegura una comunicación eficiente entre los componentes.

La aplicación se conecta con GitHub a través de Azure Static Web Apps, lo que permite integrar el repositorio directamente con el servicio de despliegue. Al configurar la Static Web App, se vincula el repositorio de GitHub y se autoriza a Azure a acceder a él, especificando la rama y la carpeta donde se encuentran los archivos del frontend. Cada vez que se realiza un *push* a la rama configurada, Azure detecta automáticamente el cambio, ejecuta el flujo de construcción y despliegue, y publica la versión más reciente de la aplicación. Este proceso implementa un flujo de CI/CD (Integración Continua y Despliegue Continuo): la Integración Continua asegura que cada cambio en el código se integre correctamente y pase por un proceso de construcción automatizado, mientras que el Despliegue Continuo garantiza que la interfaz web esté siempre sincronizada con el repositorio sin intervención manual. Esto permite un flujo de trabajo ágil, confiable y siempre actualizado.

La conexión con el repositorio de GitHub es un elemento clave del flujo de trabajo. Cada actualización del frontend desencadena un despliegue automático en Static Web Apps, mientras que el backend puede mantenerse actualizado mediante clonación o sincronización manual del repositorio. Esta estrategia combina integración continua para la interfaz con estabilidad en la ejecución de la lógica de búsqueda de imágenes, que se basa en modelos de inteligencia artificial distribuidos a través de Hugging Face, como BLIP para la generación de descripciones y CLIP para la recuperación semántica por similitud. El resultado es un sistema que procesa consultas de manera eficiente, devuelve resultados relevantes y se mantiene actualizado sin complicaciones operativas, aprovechando las capacidades de Azure para ofrecer un servicio fiable y escalable.

La aplicación web está disponible en el siguiente enlace: https://lively-glacier-085d2a210.1.azurestaticapps.net

Por motivos de costes esta aplicación solo estará disponible el día de la presentación, dos días antes y dos días después.

Figura 12. Aplicación web.

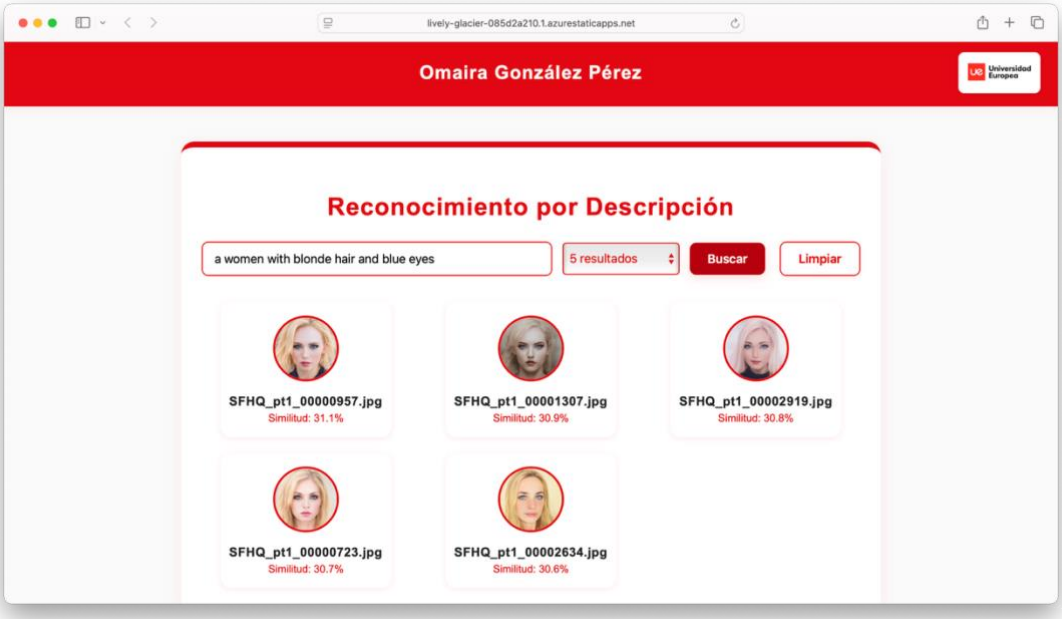

Figura 13. Aplicación web.

Se buscó que la aplicación quedara disponible en Internet de forma fiable, fácil de mantener y con un proceso de publicación claro. Para ello, se separó lo que ve el usuario (la web) de lo que hace el trabajo por detrás (el servicio que busca y devuelve resultados), aprovechando dos piezas de Azure: SWA para el frontend y una máquina virtual para el backend.

Se dejó el frontend en SWA (Static Web Apps). Se conectó el repositorio y, al subir los archivos de la web, SWA los publicó automáticamente. Desde ese momento, cada cambio que se envíe al repositorio vuelve a actualizar la web sin pasos extra.

El backend se colocó en una máquina virtual de Azure. Allí se copiaron la lógica y las imágenes, y se dejó el servicio funcionando de forma estable. Para ello, se usó Gunicorn como "motor" que ejecuta la aplicación y Nginx como "portero" que recibe las peticiones desde Internet, las encamina hacia la aplicación y gestiona la entrega de contenidos. Nginx también se encargó del HTTPS, de modo que las conexiones quedan seguras y no aparecen bloqueos en el navegador.

La máquina virtual que aloja el backend ejecuta Linux (Ubuntu 22.04) y tiene un tamaño Standard B2s, con 2 vCPU y 4 GiB de memoria RAM, lo que le permite manejar de manera eficiente las solicitudes de procesamiento de imágenes y consultas semánticas. Su arquitectura es x64 y pertenece a la generación V2, garantizando compatibilidad con las últimas actualizaciones de software y soporte de Azure.

El estado del agente de la VM es Ready, con versión 2.14.0.1, asegurando que los servicios de gestión y automatización de Azure puedan operar correctamente.

En términos de seguridad, la máquina virtual cuenta con arranque seguro y vTPM habilitado, proporcionando un entorno protegido frente a modificaciones no autorizadas durante el arranque y mejorando la integridad del sistema. La supervisión de integridad y la supervisión de mantenimiento no están activadas, lo que simplifica la administración, aunque se recomienda monitorizar estos parámetros para entornos de producción críticos.

La VM utiliza un controlador de disco SCSI, optimizado para rendimiento en operaciones de lectura/escritura, y tiene habilitada la funcionalidad de Azure de acceso puntual con la extensión enablevmAccess, facilitando la administración remota y la recuperación ante incidencias. Su configuración de red incluye una IP privada que permite comunicación interna segura dentro de la red virtual, asegurando que el backend pueda interactuar con otros servicios y con el frontend de manera confiable y estable.

Después, en el frontend se indicó que las peticiones debían ir a la dirección segura de la máquina virtual (para buscar y para mostrar imágenes). Con eso, la web en SWA habla con el backend en la nube a través de Nginx y Gunicorn, y todo queda conectado y listo para su uso.

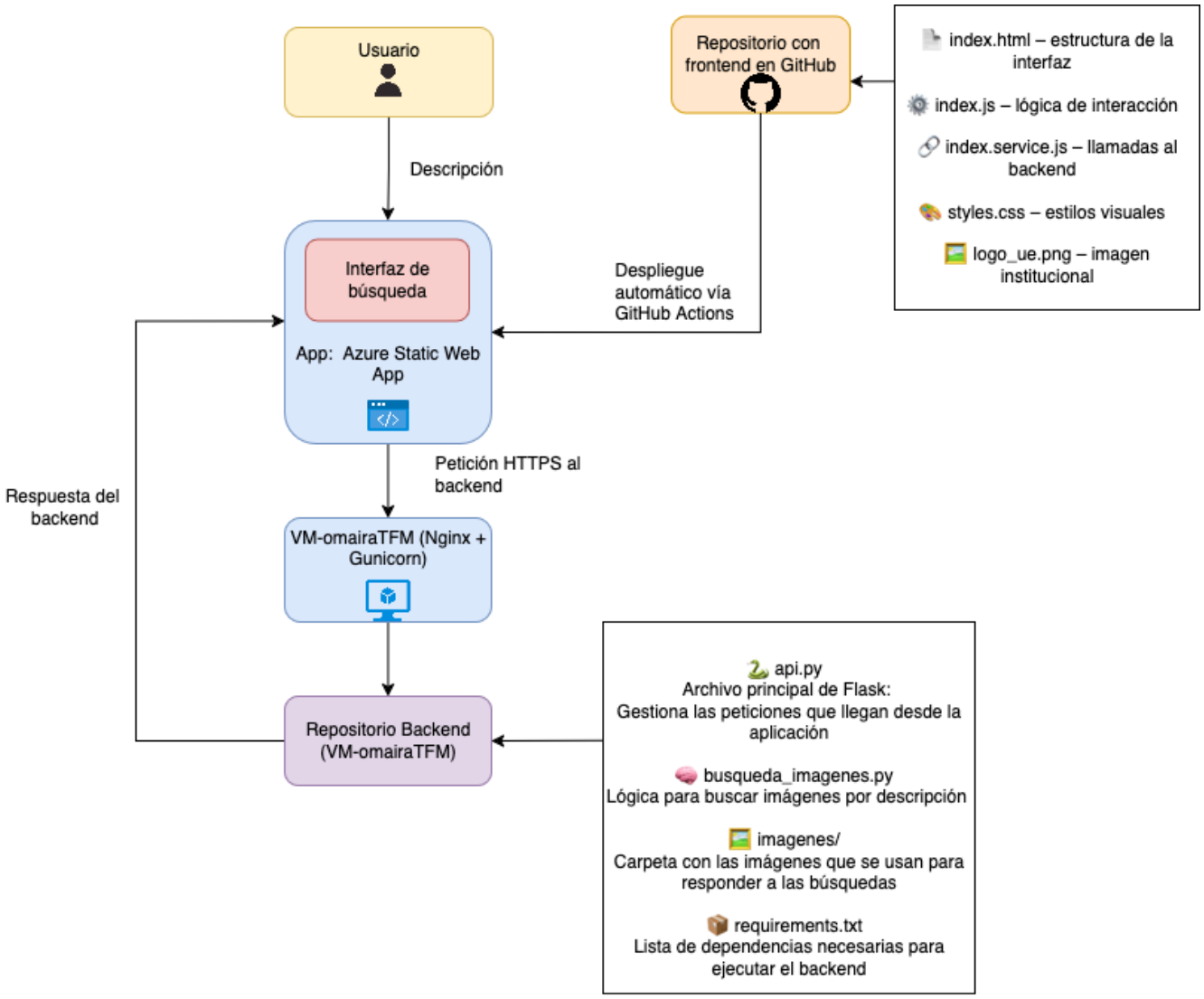

Figura 14. Arquitectura aplicación.

El diagrama muestra cómo se conectan los componentes de la aplicación en Azure: el frontend desplegado en Static Web Apps se comunica de manera segura mediante HTTPS con el backend alojado en una máquina virtual, donde Nginx y Gunicorn gestionan las solicitudes y ejecutan la lógica de búsqueda de imágenes mediante inteligencia artificial. Esta separación permite que la interfaz sea rápida y moderna, mientras que el backend procesa y devuelve resultados de forma confiable y segura.

4.5.1 Rendimiento y costes.

Durante las pruebas realizadas, el sistema de búsqueda y descripción de imágenes ha demostrado un rendimiento consistente y fiable. En la ejecución más reciente, se procesaron

550 imágenes en aproximadamente 32 segundos en entorno local, lo que equivale a una media de casi 17 imágenes por segundo. En la máquina virtual (VM) desplegada en Azure, el mismo proceso tomó 56 segundos, lo que refleja una diferencia atribuible a las condiciones del entorno de ejecución. Aun así, el sistema maneja con soltura volúmenes moderados de datos y puede entregar resultados rápidamente. Para escenarios más grandes o con demandas en tiempo real, podría ser necesario ajustar ciertos parámetros para mantener esta eficiencia.

Por otro lado, el uso de Azure ha resultado bastante económico. Los costes diarios dependen del número de ejecuciones, alcanzando 1,18 € al día. En general, los gastos diarios son bajos y permiten llevar un buen control del presupuesto. Aun así, conviene hacer un seguimiento periódico, ya que el uso prolongado podría generar aumentos importantes si no se gestionan los recursos de forma adecuada.

En resumen, el sistema combina un buen rendimiento con costes contenidos, ofreciendo una solución eficiente y sostenible para proyectos de tamaño moderado, con margen para optimizaciones futuras si se requieren volúmenes mayores o uso continuo.

Capítulo 5. CONCLUSIONES

5.1 Conclusiones del trabajo

El proyecto ha logrado cumplir su objetivo principal: facilitar la búsqueda de personas a partir de una descripción escrita mediante un sistema de reconocimiento de imágenes inteligente. La aplicación es capaz de transformar el contenido visual en descripciones automáticas, y gracias a la búsqueda semántica, identifica de manera efectiva las imágenes más relevantes según la consulta del usuario. Aunque algunas descripciones pueden resultar algo genéricas, el sistema consigue ofrecer resultados coherentes y útiles, organizando las imágenes de manera que las más relevantes aparecen primero. Este comportamiento refleja la efectividad del enfoque basado en inteligencia artificial y demuestra que es posible combinar precisión técnica con utilidad práctica para el usuario final.

Además, el despliegue de la aplicación en Azure ha facilitado que el sistema sea accesible desde cualquier red, seguro y sencillo de mantener. El frontend está alojado en Static Web Apps y se actualiza automáticamente con los cambios realizados en GitHub, asegurando que la interfaz siempre refleje el código más reciente. Por su parte, el backend funciona en una máquina virtual protegida, donde procesa las solicitudes de manera eficiente, gestionando tanto la lógica de búsqueda como la recuperación de imágenes mediante los modelos de inteligencia artificial. Este esquema permite separar de manera clara la presentación de la información de su procesamiento, garantizando estabilidad, seguridad y escalabilidad.

En conjunto, los resultados obtenidos muestran que la aplicación cumple no solo con los objetivos técnicos de precisión y confiabilidad, sino que también ofrece una experiencia de usuario fluida y un flujo de trabajo ágil para los desarrolladores. La integración de inteligencia artificial con un despliegue en la nube seguro y una actualización automática del frontend permite que la solución sea práctica, escalable y fácilmente mantenible, consolidando un sistema completo que combina innovación tecnológica con utilidad real para los usuarios.

5.2 Conclusiones personales

El desarrollo de este proyecto ha sido una experiencia muy enriquecedora y formativa. Trabajar en la creación de un sistema que combina inteligencia artificial con una interfaz web accesible me permitió profundizar en áreas como procesamiento de imágenes, modelos de lenguaje y embeddings semánticos, así como en el despliegue de aplicaciones en la nube. Cada fase del proyecto, supuso un desafío que me enseñó a integrar de manera práctica teoría y tecnología.

Además, aprender a desplegar la aplicación en Azure y a conectar el frontend con GitHub me brindó una comprensión real del flujo de trabajo moderno de desarrollo y despliegue, incluyendo aspectos de seguridad, escalabilidad y automatización. Personalmente, este proyecto me permitió ver cómo la inteligencia artificial puede aplicarse para resolver

problemas concretos de manera tangible y útil, y me motivó a seguir explorando soluciones innovadoras en el ámbito de la visión y lenguaje.

Creo que este proyecto también tiene un valor para otros, ya que demuestra cómo la tecnología puede facilitar tareas complejas como la búsqueda visual, haciendo que información útil esté al alcance de cualquier usuario. En conjunto, no solo he adquirido conocimientos técnicos valiosos, sino que también he ganado confianza en mi capacidad para desarrollar soluciones integrales que combinan inteligencia artificial, desarrollo web y servicios en la nube.

Capítulo 6. FUTURAS LÍNEAS DE TRABAJO.

A medida que el sistema evoluciona, una de las líneas más ambiciosas sería convertirlo en una solución capaz de gestionar grandes volúmenes de imágenes, como los que se generan diariamente en redes sociales, medios digitales, plataformas educativas o incluso en entornos científicos. Para ello, sería necesario adoptar una arquitectura basada en Big Data, que permita procesar, analizar y almacenar millones de imágenes de forma eficiente. Tecnologías como Apache Spark, Apache Flink o Databricks ofrecen entornos distribuidos que permiten dividir el trabajo en tareas paralelas, acelerando el procesamiento sin comprometer la calidad de los resultados.

Además, el uso de plataformas como Google BigQuery, Azure Synapse Analytics o Amazon Redshift permitiría realizar análisis masivos sobre los datos visuales procesados. Esto abriría la puerta a estudios más profundos sobre patrones visuales, tendencias de contenido, o incluso análisis socioculturales basados en imágenes. Por ejemplo, se podrían detectar qué tipos de escenas predominan en ciertas regiones, cómo varía la representación visual de conceptos según el idioma, o qué elementos aparecen con mayor frecuencia en contextos educativos frente a comerciales.

Otra línea de trabajo relevante sería incorporar sistemas de aprendizaje automático que se actualicen de forma continua. Herramientas como TensorFlow Extended (TFX) o MLflow permiten construir pipelines que se alimentan de nuevos datos y ajustan los modelos automáticamente. Esto haría que el sistema no solo describa imágenes, sino que aprenda de ellas con el tiempo, adaptándose a nuevas formas de expresión visual, estilos emergentes o necesidades específicas de los usuarios.

Una evolución interesante del proyecto sería incorporar sistemas de almacenamiento distribuido que permitan guardar y acceder a millones de imágenes de forma rápida y segura. Tecnologías como Hadoop Distributed File System (HDFS) o Ceph ofrecen soluciones escalables para manejar grandes volúmenes de datos visuales sin perder rendimiento. Además, se podrían aplicar técnicas de indexación visual usando herramientas como FAISS (Facebook AI Similarity Search), que permiten buscar imágenes similares basándose en características visuales, no solo en texto. Esto abriría la puerta a funcionalidades como “buscar por imagen” o “encontrar contenido visual relacionado”, muy útiles en sectores como el comercio electrónico o la educación.

En paralelo al procesamiento, es fundamental pensar cómo se almacenan y organizan las imágenes y sus descripciones. El objetivo no es solo guardar datos, sino convertirlos en conocimiento accesible y reutilizable. Para ello, se podrían utilizar bases de datos no relacionales como MongoDB, Cassandra o Amazon DynamoDB, que permiten almacenar imágenes junto con sus metadatos de forma flexible y escalable. Estas tecnologías facilitan búsquedas rápidas y permiten estructurar la información visual de manera que sea útil para distintos perfiles de usuario: investigadores, docentes, diseñadores, archivistas, etc.

Para facilitar el trabajo colaborativo, se podría integrar el sistema con plataformas en la nube como Azure Blob Storage, Google Cloud Storage o Amazon S3, que permiten compartir y sincronizar datos entre distintos usuarios o dispositivos. Esto abriría la puerta a proyectos colectivos donde varios equipos puedan trabajar sobre el mismo conjunto de imágenes, añadir anotaciones, corregir descripciones o construir colecciones temáticas. Además, el uso de tecnologías como ElasticSearch permitiría crear motores de búsqueda visuales avanzados, capaces de responder a consultas complejas como “imágenes con tonos cálidos y personas en movimiento” o “escenarios urbanos al atardecer”.

Finalmente, se podría desarrollar una interfaz visual atractiva y funcional, utilizando frameworks como React o Angular, que permita explorar las imágenes como si fueran parte de una galería inteligente. Esta interfaz no solo facilitaría la navegación, sino que también podría incluir filtros dinámicos, recomendaciones basadas en intereses previos, y opciones de exportación para distintos formatos. En conjunto, estas mejoras convertirían el sistema en una plataforma robusta, escalable y útil para una amplia variedad de aplicaciones reales.

Capítulo 7. REFERENCIAS

[1] T. B. Brown et al., «Language Models are Few-Shot Learners», *arXiv*, 2020. [En línea]. Disponible en: https://arxiv.org/abs/2005.14165

[2] OpenAI, «GPT-4», *OpenAI Research*, 14 de marzo de 2023. [En línea]. Disponible en: https://openai.com/research/gpt-4

[3] Anthropic, «Claude 3», *Anthropic*, 4 de marzo de 2024. [En línea]. Disponible en: https://www.anthropic.com/index/claude-3

[4] J. Devlin, M.-W. Chang, K. Lee y K. Toutanova, «BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding», *arXiv*, 2018. [En línea]. Disponible en: https://arxiv.org/abs/1810.04805

[5] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger y I. Sutskever, «Learning Transferable Visual Models From Natural Language Supervision», *arXiv*, 26 de febrero de 2021. [En línea]. Disponible en: https://arxiv.org/abs/2103.00020

[6] J. Li, D. Li, C. Xiong y S. Hoi, «BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation», *arXiv*, 15 de febrero de 2022. [En línea]. Disponible en: https://arxiv.org/abs/2201.12086

[7] H. Liu, C. Li, Q. Wu y Y. J. Lee, «Visual Instruction Tuning», *arXiv*, 11 de diciembre de 2023. [En línea]. Disponible en: https://arxiv.org/abs/2304.08485

[8] Parlamento Europeo y Consejo de la Unión Europea, «Reglamento (UE) 2016/679 del Parlamento Europeo y del Consejo, de 27 de abril de 2016, relativo a la protección de las personas físicas en lo que respecta al tratamiento de datos personales y a la libre circulación de estos datos», Diario Oficial de la Unión Europea, L 119, 4 de mayo de 2016. [En línea]. Disponible en: https://eur-lex.europa.eu/legal-content/ES/TXT/?uri=CELEX%3A32016R0679

[9] Comisión Europea, «Propuesta de Reglamento del Parlamento Europeo y del Consejo por el que se establecen normas armonizadas sobre inteligencia artificial (Ley de Inteligencia Artificial) y se modifican ciertos actos legislativos de la Unión», COM(2021) 206 final, 21 de abril de 2021. [En línea]. Disponible en: https://eur-lex.europa.eu/legal-content/ES/TXT/?uri=CELEX%3A52021PC0206

[10] Hugging Face, «Vision Language Models Explained», 11 de abril de 2024. [En línea]. Disponible en: https://huggingface.co/blog/vlms

[11] David Beniaguev, «Synthetic Faces High Quality (SFHQ) part 1», *Kaggle*, 2022. [En línea]. Disponible en: https://www.kaggle.com/datasets/selfishgene/synthetic-faces-high-quality-sfhq-part-1

[12] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger y I. Sutskever, «Learning Transferable Visual Models From Natural Language Supervision», *arXiv*, 26 de febrero de 2021. [En línea]. Disponible en: https://arxiv.org/abs/2103.00020

[13] J. Li, D. Li, C. Xiong y S. Hoi, «BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation», *arXiv*, 28 de enero de 2022. [En línea]. Disponible en: https://arxiv.org/abs/2201.12086

[14] N. Reimers y I. Gurevych, «Sentence Embeddings using Siamese BERT-Networks», *arXiv*, 27 de agosto de 2019. [En línea]. Disponible en: https://arxiv.org/abs/1908.10084

[15] OpenAI, «CLIP (Contrastive Language-Image Pretraining)», *GitHub*, 2021. [En línea]. Disponible en: https://github.com/openai/CLIP

[16] Salesforce, «BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation», *GitHub*, 2022. [En línea]. Disponible en: https://github.com/salesforce/BLIP

[17] Hugging Face, «Transformers Documentation», *Hugging Face*, 2023. [En línea]. Disponible en: https://huggingface.co/docs/transformers/index

[18] Scikit-learn, «cosine_similarity», *Scikit-learn Documentation*, 2023. [En línea]. Disponible en: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.pairwise.cosine_similarity.html

[19] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser y I. Polosukhin, «Attention is All You Need», *arXiv*, 6 de octubre de 2019. [En línea]. Disponible en: https://arxiv.org/abs/1910.03771

[20] J. C.-C. Chou y N. Alam, «Embedding Geometries of Contrastive Language-Image Pre-Training», *arXiv*, 19 de septiembre de 2024. [En línea]. Disponible en: https://arxiv.org/pdf/2409.13079

[21] GitHub, «Omaira14/omaira_tfm: generar_descripciones.py», 2025. [En línea]. Disponible en: https://github.com/Omaira14/omaira_tfm/blob/main/generar_descripciones.py

[22] GitHub, «Omaira14/omaira_tfm: busqueda_imagnes.py», 2025. [En línea]. Disponible en: https://github.com/Omaira14/omaira_tfm/blob/main/busqueda_imagnes.py

[23] A. Radford et al., «Learning Transferable Visual Models From Natural Language Supervision», *arXiv*, 26 de febrero de 2021. [En línea]. Disponible en: https://arxiv.org/abs/2103.00020

[24] Comisión Europea, «Proposal for a Regulation of the European Parliament and of the Council laying down harmonised rules on artificial intelligence (Artificial Intelligence Act) and

amending certain Union legislative acts», 2021. [En línea]. Disponible en: https://eur-lex.europa.eu/legal-content/ES/TXT/?uri=CELEX%3A52021PC0206