IES PERE MARIA ORTS

ESO, Batxillerat, Arts, Formació Professional

Conselleria de Educación, Cultura, Universidades y Empleo

Curso de Especialización en Inteligencia Artificial y Big Data

Modelos de datos y Sistemas de Aprendizaje Automático

Reconocimiento labial con MediaPipe y Redes Neuronales

Trabajo fin de estudio presentado por:	Raúl Márquez Roig
Director/a:	David Campoy Miñarro
Fecha:	26/05/2025
Repositorio del código fuente:	https://github.com/raulmarquez93/Lenguaje_labial

Resumen

Este proyecto final tiene como objetivo desarrollar un prototipo funcional capaz de reconocer palabras mediante la lectura de labios, utilizando herramientas de inteligencia artificial (IA) y visión por ordenador. El problema de partida se centra en la complejidad de interpretar correctamente los movimientos labiales de una persona para identificar qué palabra está pronunciando, una tarea que requiere captar tanto la forma estática de los labios como su dinámica temporal. La innovación de este trabajo reside en la implementación de redes neuronales aplicadas al análisis de secuencias visuales, combinando técnicas modernas de aprendizaje profundo con herramientas avanzadas de procesamiento de imágenes como MediaPipe.

Para abordar este desafío, se han diseñado y entrenado varios modelos de IA, incluyendo redes neuronales densas y redes neuronales recurrentes del tipo LSTM (Long Short-Term Memory), específicamente adaptadas para analizar las secuencias temporales que caracterizan el movimiento de los labios al hablar. El sistema se ha alimentado con datos propios y ha sido evaluado en términos de precisión y eficacia para reconocer al menos tres palabras diferentes.

Las principales conclusiones del trabajo revelan que la combinación de visión e inteligencia artificiales permite construir prototipos efectivos de lectura automática de labios, sentando las bases para futuros sistemas capaces de manejar vocabularios mucho más amplios. Este enfoque no solo captura la información visual relevante, sino que también abre la puerta a aplicaciones inclusivas, interfaces silenciosas y asistentes virtuales más avanzados.

Palabras clave: Inteligencia Artificial, lectura de labios, MediaPipe, redes neuronales, LSTM

Abstract

This final project aims to develop a functional prototype capable of recognizing words through lip reading, using artificial intelligence (AI) and computer vision tools. The core challenge lies in the complexity of correctly interpreting a person's lip movements to identify which word is being spoken — a task that requires capturing both the static shape of the lips and their temporal dynamics. The innovation of this work resides in the implementation of neural networks applied to the analysis of visual sequences, combining modern deep learning techniques with advanced image processing tools such as MediaPipe.

To address this challenge, several AI models have been designed and trained, including dense neural networks and recurrent neural networks of the LSTM (Long Short-Term Memory) type, specifically adapted to analyze the temporal sequences that characterize lip movements during speech. The system has been trained on self-collected data and has been evaluated in terms of accuracy and effectiveness in recognizing at least three different words.

The main conclusions of this work reveal that the combination of computer vision and artificial intelligence enables the construction of effective prototypes for automatic lip reading, laying the groundwork for future systems capable of handling much larger vocabularies. This approach not only captures relevant visual information but also opens the door to inclusive applications, silent interfaces, and more advanced virtual assistants.

Keywords: Artificial Intelligence, lip reading, MediaPipe, neural networks, LSTM.

Índice de contenidos

Índice de figuras

Índice de tablas

1. Introducción

El presente trabajo describe el desarrollo de un prototipo funcional de reconocimiento de palabras a partir de la lectura de labios, utilizando herramientas de visión artificial como MediaPipe y redes neuronales. Este proyecto se ha desarrollado como práctica final para consolidar los conocimientos adquiridos durante el curso de Inteligencia Artificial.

Tiene como propósito integrar tecnologías actuales de deep learning con un enfoque práctico que permita el reconocimiento de lenguaje no verbal a partir de secuencias visuales. El sistema está orientado a facilitar la comunicación silenciosa, apoyar a personas con discapacidades auditivas y permitir interacciones más avanzadas con asistentes virtuales.

A lo largo de este documento se exponen las motivaciones del proyecto, el contexto tecnológico en el que se enmarca, los objetivos generales y específicos planteados, la metodología utilizada, el desarrollo del sistema, los resultados obtenidos y las futuras líneas de trabajo.

1.1. Motivación

La motivación que dio origen a este proyecto surge de una combinación entre la sensibilidad social, la innovación tecnológica y el atractivo de resolver un reto técnico significativo. En primer lugar, se reconoce la utilidad social de un sistema que permita comprender el lenguaje no verbal. Muchas personas con discapacidades auditivas podrían beneficiarse de tecnologías que interpreten movimientos labiales, especialmente en situaciones donde los medios tradicionales de comunicación resultan ineficaces o excluyentes.

Desde el punto de vista tecnológico, el reconocimiento visual de palabras tiene una gran cantidad de aplicaciones. No solo puede implementarse en asistentes virtuales que funcionen en silencio o en ambientes ruidosos, sino también en dispositivos portátiles como teléfonos móviles, gafas inteligentes, o sistemas embebidos en robots de asistencia. Estas capacidades abren un abanico de posibilidades para su uso tanto en el entorno doméstico como en ámbitos profesionales, sanitarios o educativos.

Además, este proyecto presenta un reto técnico apasionante. Leer los labios a partir de una secuencia de video implica trabajar con múltiples disciplinas: visión por computador para detectar rostros y labios, aprendizaje profundo para modelar patrones visuales complejos y redes neuronales temporales como LSTM para entender la evolución en el tiempo de esos patrones. Enfrentarse a este reto ha supuesto no solo aplicar lo aprendido en el curso, sino también profundizar en nuevas áreas de conocimiento, investigar soluciones reales y hacer ajustes constantes durante el desarrollo del prototipo.

1.2. Planteamiento del trabajo

El objetivo principal de este trabajo es desarrollar un sistema que, utilizando la cámara de un dispositivo, pueda reconocer al menos dos o tres palabras basándose exclusivamente en el movimiento de los labios, sin usar audio. Este prototipo servirá como base para, en un futuro, ampliar el sistema a un vocabulario mayor.

Para lograr este objetivo general, se identifican varios requerimientos esenciales. El sistema debe ser capaz de funcionar con videos grabados previamente. La predicción debe basarse en secuencias de imágenes, extrayendo patrones visuales en movimiento y no solamente en fotogramas aislados. Esto implica diseñar una arquitectura de red neuronal que sea sensible a la dinámica temporal.

Otro punto fundamental es que el modelo debe ser entrenado con datos completamente originales, lo cual requiere grabar y preparar un conjunto de datos personalizado. De igual manera, la eficacia del sistema debe ser cuantificable a través de métricas objetivas como la precisión, la sensibilidad o el F1-score. Finalmente, se ha considerado desde el inicio que este prototipo debe permitir una futura ampliación a un vocabulario más amplio, apuntando a la posibilidad de reconocer hasta 100 palabras frecuentes del lenguaje cotidiano.

Este prototipo funcional se estructura como una prueba de concepto, pero con bases técnicas suficientemente sólidas como para evolucionar hacia sistemas comerciales o industriales en el corto y mediano plazo.

1.3. Estructura del trabajo

Este documento se organiza en varios capítulos. En el primero se presenta la introducción general al problema, donde se contextualiza el desarrollo del sistema de reconocimiento labial dentro del campo de la inteligencia artificial aplicada. Además, se exponen las motivaciones que impulsaron el proyecto y se identifican tanto los objetivos generales como los desafíos iniciales, incluyendo una sección específica dedicada a las dificultades encontradas durante las fases de preparación, grabación de datos y entrenamiento del modelo.

En el segundo capítulo se aborda el contexto y los antecedentes tecnológicos más relevantes. Aquí se profundiza en el funcionamiento de herramientas como MediaPipe para la detección de puntos faciales, así como en los principios de redes neuronales profundas, con especial atención a las arquitecturas recurrentes como LSTM. También se recogen ejemplos de proyectos previos de lectura de labios, tanto en entornos académicos como industriales, para establecer comparaciones y destacar las particularidades de la presente propuesta.

El tercer capítulo se dedica a la formulación detallada de los objetivos específicos y a la explicación meticulosa de la metodología adoptada. Se explican los criterios seguidos para la creación del dataset, el diseño del flujo de procesamiento de datos, las herramientas empleadas en la transformación y aumento del material audiovisual, y las decisiones tomadas durante la fase de implementación del modelo de aprendizaje.

En el cuarto capítulo se expone en profundidad el proceso de desarrollo del prototipo. Aquí se describe paso a paso cómo se llevó a cabo la grabación automatizada de videos, la segmentación de los mismos por clase, la aplicación de técnicas de aumento de datos con Albumentations, la extracción de frames y la detección y recorte de labios mediante MediaPipe. Además, se detalla la estructura del modelo LSTM, los parámetros utilizados en su entrenamiento y los ajustes realizados para mejorar su rendimiento.

El quinto capítulo está centrado en la evaluación del sistema. Se presentan las métricas de precisión, pérdida y exactitud, y se analizan los resultados obtenidos tanto en entrenamiento como en validación. Se incluyen reflexiones sobre la robustez del modelo ante transformaciones, errores comunes en la clasificación y posibles causas de confusión entre clases.

Por último, el sexto capítulo recopila las principales conclusiones derivadas del trabajo, reflexionando sobre los logros alcanzados, las limitaciones técnicas detectadas y el valor formativo de la experiencia. Asimismo, se plantean propuestas concretas para futuras mejoras, incluyendo la expansión del vocabulario, la inclusión de múltiples hablantes y la implementación de modelos más avanzados o combinados.

El propósito de esta organización es guiar al lector a través de todas las fases del proyecto, ofreciendo una visión integral del proceso de construcción del sistema, desde sus fundamentos teóricos hasta su validación empírica, de forma clara, argumentada y coherente. para ofrecer una comprensión completa y detallada del proceso seguido, sirviendo tanto de memoria técnica como de base para posibles continuaciones del proyecto.

2. Contexto y Estado del Arte

Antes de abordar el desarrollo del prototipo, es necesario entender las herramientas tecnológicas y los avances recientes relacionados con la lectura de labios automática. En este capítulo se revisan conceptos como los modelos de redes neuronales, el papel de MediaPipe en la extracción de landmarks faciales y el uso de Redes Neuronales para generalizar.

Estos elementos no solo representan el corazón técnico del proyecto, sino que además reflejan un estado del arte en constante evolución, donde convergen múltiples disciplinas: visión por computador, aprendizaje automático y procesamiento de señales visuales. A través del análisis de estos componentes, se establece el fundamento teórico y práctico que sostiene la arquitectura del sistema diseñado. También se detallarán las estrategias empleadas para procesar la información capturada por la cámara y transformarla en datos relevantes para el

aprendizaje, junto con una revisión de proyectos similares que han influido o servido como referencia.

Este marco contextual permitirá comprender por qué ciertas decisiones técnicas fueron adoptadas y cómo se relacionan con soluciones existentes en la literatura o la industria. Se busca con ello no solo justificar el enfoque propuesto, sino también identificar oportunidades de mejora y crecimiento para versiones futuras del prototipo.

2.1. MediaPipe y la visión por computador aplicada a labios

MediaPipe es un framework de código abierto desarrollado por Google que permite realizar tareas complejas de visión por computador en tiempo real, con un enfoque modular y multiplataforma. Una de sus características más destacadas es su capacidad para detectar y seguir puntos clave (landmarks) del rostro humano mediante el modelo conocido como Face Mesh. Este modelo predice con alta precisión la ubicación de 468 puntos faciales tridimensionales utilizando únicamente imágenes RGB, sin necesidad de sensores de profundidad.

En el contexto del reconocimiento labial, esta tecnología resulta particularmente útil porque permite identificar con precisión la región de los labios y su contorno, incluso cuando el sujeto realiza movimientos sutiles o está en movimiento. El módulo de Face Mesh no solo proporciona coordenadas bidimensionales, sino que también estima la profundidad relativa de cada punto, lo cual mejora la robustez del sistema frente a variaciones de iluminación o perspectiva.

MediaPipe ofrece una arquitectura altamente optimizada, pensada para ejecutarse eficientemente incluso en dispositivos móviles. Este enfoque centrado en la eficiencia permite que soluciones avanzadas de visión artificial, como la lectura de labios, puedan ser implementadas fuera del laboratorio, en entornos reales y con hardware limitado. En este proyecto, se ha utilizado Face Mesh con una configuración específica que refina la detección

de landmarks alrededor de la boca, permitiendo recortar dinámicamente la región de interés en cada frame del video y alimentarla al sistema de clasificación.

Además de Face Mesh, MediaPipe también ofrece soluciones para pose detection, hand tracking, objectron (detección 3D de objetos), entre otras, todas integradas bajo un diseño de gráficos de procesamiento configurable, donde los desarrolladores pueden conectar módulos como si se tratara de una red de nodos. Este nivel de flexibilidad permite adaptar el sistema a distintos casos de uso sin necesidad de reconstruir completamente el flujo de datos.

En este proyecto, se aprovechó esta capacidad modular para extraer regiones labiales uniformes de videos aumentados, asegurando que el modelo de red neuronal recibiera entradas consistentes en cuanto a tamaño, escala y orientación, mejorando así la precisión del aprendizaje supervisado posterior.

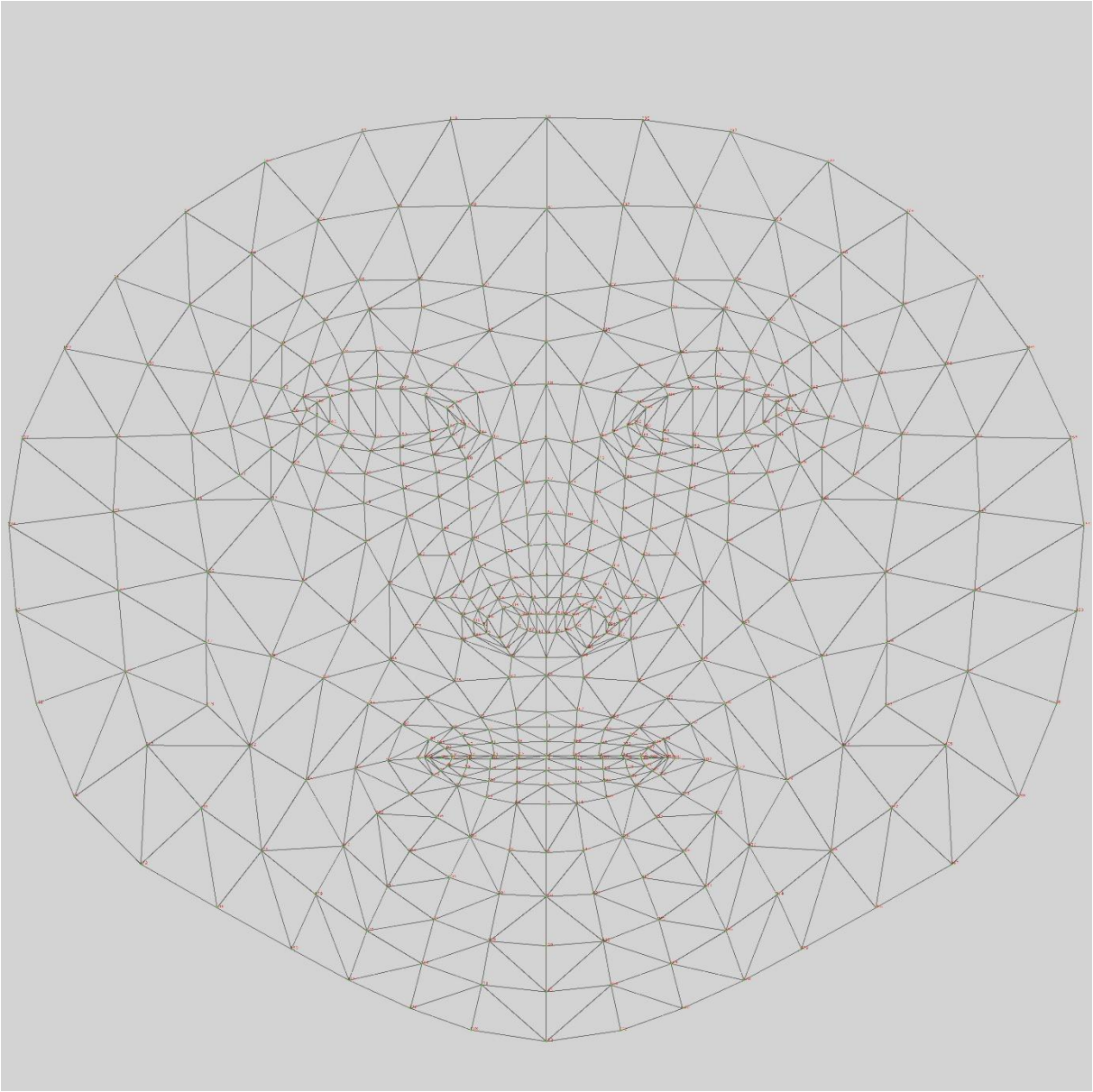


Figura 1. Diagrama del rostro con los 468 puntos de MediaPipe

2.2.Redes neuronales y aprendizaje profundo

En el corazón del sistema propuesto se encuentra una red neuronal que combina procesamiento espacial y secuencial, haciendo uso tanto de convoluciones bidimensionales como de unidades LSTM (Long Short-Term Memory). Para entender este modelo, es fundamental repasar los conceptos básicos que lo componen.

Una red neuronal es un modelo computacional inspirado en la estructura del cerebro humano. Está conformada por capas de neuronas artificiales conectadas entre sí. Las más comunes incluyen las capas densas o completamente conectadas, que calculan combinaciones lineales de las entradas seguidas de funciones de activación no lineales. Sin embargo, en el contexto del procesamiento de imágenes y vídeo, este tipo de capas no es suficiente por sí solo, ya que no tienen en cuenta la estructura espacial de los datos. Al tratar cada píxel como un valor independiente, las capas densas pierden la relación entre píxeles vecinos, lo que dificulta la detección de patrones visuales como bordes, formas o movimientos. Además, aplicar capas densas directamente a imágenes de alta resolución implica una cantidad extremadamente elevada de parámetros, lo que vuelve el modelo ineficiente y propenso al sobreajuste. Por estas razones, en tareas visuales complejas se recurre a arquitecturas especializadas, como las redes convolucionales (CNN), que permiten analizar la información visual respetando su estructura espacial.

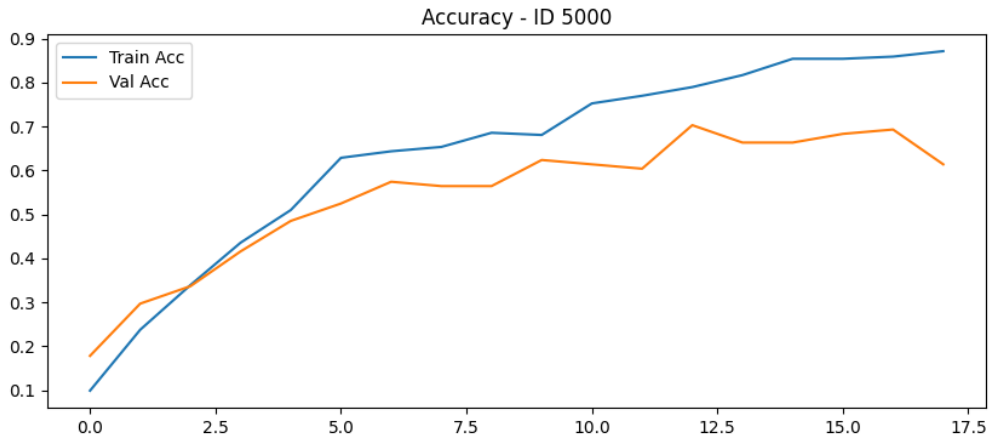


Figura 2. Curva de aprendizaje durante el entrenamiento.

2.3. Trabajos y estudios previos de lectura de labios

El campo del reconocimiento visual de palabras ha crecido significativamente en los últimos años, impulsado por avances en visión por computador y aprendizaje profundo. A pesar de este progreso, aún existen desafíos relacionados con la variabilidad lingüística, las condiciones del entorno y la disponibilidad de datos. En este apartado se exponen dos estudios recientes de alto impacto que aportan referentes concretos y validados sobre cómo abordar la lectura automática de labios.

Uno de los trabajos más relevantes es el llevado a cabo por el equipo de la Universidad de Teherán en el contexto del robot humanoide Surena-V. Este estudio se centró en integrar un sistema de lectura de labios en el robot, permitiéndole interpretar órdenes sin necesidad de audio. Para lograrlo, exploraron dos métodos complementarios: un enfoque indirecto basado en el seguimiento de landmarks faciales alrededor de los labios, y otro directo, fundamentado en la utilización de redes convolucionales (CNN) y LSTM aplicadas directamente sobre las secuencias de vídeo. El modelo que mostró mejor desempeño fue el LSTM, alcanzando una precisión del 89% y siendo finalmente implementado en el robot para tareas reales de interacción humano-máquina. El trabajo destaca la importancia de combinar estructuras de red especializadas con un dataset propio, adaptado a las palabras clave necesarias para el contexto del robot. También hace énfasis en los retos técnicos encontrados, como las dificultades con movimientos bruscos o el fallo del detector labial en ciertas posiciones del rostro.

Un segundo estudio de gran valor es el realizado por investigadores filipinos, quienes desarrollaron un modelo de lectura de labios específicamente orientado al idioma tagalo. Este modelo se construyó con una arquitectura híbrida CNN-LSTM inspirada en LipNet, pero con la particularidad de adoptar un enfoque multimodal que integra tanto datos visuales como información textual. El sistema procesa los movimientos labiales extraídos de los vídeos mientras simultáneamente analiza el contexto lingüístico de las palabras esperadas. Este diseño logró no solo una alta precisión en la validación (89.5%) sino también un aumento del 25% en la velocidad de procesamiento, al evitar componentes acústicos. Su estrategia de preprocesamiento que incluye la extracción precisa de labios, normalización de imágenes y tokenización textual fue clave para alcanzar estos resultados. Además, este trabajo pone de relieve la importancia de atender idiomas poco representados en la literatura y demuestra que modelos bien adaptados pueden ser efectivos incluso con recursos limitados

Ambos estudios respaldan y validan el enfoque adoptado en el presente proyecto. La combinación de CNN para procesamiento espacial y LSTM para modelado secuencial, junto con un cuidadoso diseño del dataset y un proceso de preprocesamiento robusto, se alinea con las mejores prácticas identificadas en la literatura. Asimismo, ofrecen perspectivas útiles para futuras ampliaciones, como la integración de datos textuales, la generalización a múltiples hablantes o el uso de cámaras 3D para captar profundidad labial.

3. Objetivos y metodología de trabajo

El desarrollo de este prototipo se ha guiado por una serie de objetivos definidos y una metodología estructurada en varias fases. A partir del análisis de los desafíos actuales en lectura automática de labios, se planteó la creación de un sistema funcional que aproveche las capacidades de la visión por computador y las redes neuronales temporales. Esta sección expone tanto el objetivo general que motivó todo el diseño como los objetivos específicos que permitieron avanzar de forma ordenada. Asimismo, se detalla la metodología de trabajo

empleada, en la que se combinan herramientas técnicas, decisiones estratégicas y procesos experimentales orientados a la validación del sistema.

3.1. Objetivo general

El objetivo principal de este proyecto ha sido diseñar e implementar un prototipo funcional capaz de reconocer entre dos y tres palabras. Aunque lo he ampliado a 10 palabras (en este caso, números del 1 al 10) únicamente a partir de los movimientos de los labios, sin necesidad de utilizar audio. Para lograrlo, se ha hecho uso de secuencias de vídeo y se han aplicado técnicas de procesamiento visual con redes neuronales profundas, específicamente una arquitectura híbrida de tipo CNN-LSTM. El sistema debe ser escalable y adaptable para futuras ampliaciones, como el reconocimiento de un vocabulario mayor y su implementación en dispositivos de uso cotidiano.

3.2. Objetivos específicos

Para traducir el objetivo general en acciones concretas y medibles, se definieron varios objetivos específicos que han guiado el desarrollo del proyecto desde sus primeras fases hasta su etapa de validación. Estos objetivos no sólo describen tareas técnicas, sino que están pensados para estructurar progresivamente un sistema robusto, escalable y adaptado al contexto de uso propuesto.

El primer gran objetivo fue establecer un mecanismo fiable y automatizado de recolección de datos. Para ello, se planteó el diseño de una aplicación web capaz de guiar visualmente al usuario durante la grabación de vídeos, sincronizando un temporizador con la pronunciación de secuencias numéricas. Esta aplicación no sólo debía facilitar el proceso de captura, sino también generar automáticamente los nombres de archivo y etiquetas necesarias para su uso posterior, evitando así errores humanos en el etiquetado manual.

Una vez asegurado un entorno controlado de grabación, el siguiente paso consistió en generar un conjunto de datos propio. Este dataset debía contener suficientes muestras por cada clase (números del 1 al 10), grabadas bajo condiciones similares y con una estructura uniforme. La creación de este conjunto no fue un simple proceso de acumulación de vídeos, sino una

operación cuidadosa que implicó segmentación precisa, estandarización de formato y validación manual de la calidad de cada muestra.

Dado que los datos originales no resultaban suficientes para entrenar un modelo con capacidad de generalización, se estableció como objetivo implementar técnicas avanzadas de aumento de datos. Mediante transformaciones visuales aleatorias —como rotación, inversión horizontal, variación de brillo y desenfoque— se logró multiplicar artificialmente la cantidad y variedad de ejemplos disponibles, simulando condiciones reales más diversas sin necesidad de capturar nuevos vídeos.

En paralelo, se definió como objetivo la extracción precisa de la región labial en cada fotograma. Esta tarea, crítica para reducir el ruido en los datos de entrada, se abordó mediante el uso de MediaPipe Face Mesh, que permite identificar con alta precisión los puntos clave del rostro. A partir de esta detección se recortó dinámicamente la zona de los labios, se transformó a escala de grises y se normalizó para generar arrays compatibles con redes neuronales.

Sobre la base de estos datos preprocesados, se planteó como objetivo fundamental el diseño e implementación de un modelo de red neuronal híbrido CNN-LSTM. Las capas convolucionales debían encargarse de extraer información espacial de cada imagen, mientras que las capas LSTM se encargarían de capturar las dependencias temporales a lo largo de la secuencia. Este enfoque se seleccionó por su eficacia comprobada en tareas que combinan procesamiento de imágenes con reconocimiento de patrones secuenciales, como es el caso de la lectura labial.

Finalmente, y no menos importante, se estableció la necesidad de evaluar rigurosamente el desempeño del modelo. Esto incluyó no solo calcular métricas tradicionales como precisión y pérdida, sino también analizar errores de clasificación, revisar matrices de confusión y comparar distintas configuraciones del modelo. Estas evaluaciones permiten identificar debilidades específicas, afinar parámetros y validar la viabilidad del sistema en contextos más amplios.

3.3. Metodología de trabajo

La metodología adoptada para este proyecto se basa en un enfoque experimental y modular, con una progresión iterativa desde la generación de datos hasta el entrenamiento del modelo.

Todo comenzó con la **recolección de datos**, donde se desarrolló una aplicación web interactiva para guiar la grabación de videos con cámara web. Esta herramienta permitió automatizar tanto la captura como el etiquetado, sincronizando el vídeo con un contador que indicaba visualmente qué número debía pronunciarse. Cada grabación se realizó en formato .webm y se almacenó localmente.

Posteriormente, se llevó a cabo la **conversión y segmentación** de los vídeos utilizando herramientas basadas en FFMPEG. Los clips fueron divididos en fragmentos de 2 segundos y clasificados automáticamente en carpetas según el número correspondiente.

El siguiente paso consistió en aplicar **técnicas de aumento de datos**, utilizando la librería Albumentations. Se generaron cinco variantes de cada vídeo aplicando transformaciones aleatorias como rotación, brillo, desenfoque y compresión. Esto amplió considerablemente el volumen del dataset, mejorando su diversidad visual.

Luego se procedió a la **extracción de frames**, donde cada vídeo fue descompuesto en imágenes individuales. Cada secuencia de vídeo se convirtió en una carpeta con 30 fotogramas ordenados cronológicamente. A continuación, se utilizó MediaPipe Face Mesh para identificar los landmarks labiales y recortar dinámicamente la región de interés. Estas regiones fueron normalizadas y almacenadas como arrays NumPy, con dimensión fija y en escala de grises.

Con el dataset ya preparado, se diseñó y entrenó un modelo basado en una arquitectura **CNN-LSTM**. Las capas convolucionales se aplicaron a cada frame individualmente (mediante TimeDistributed) para extraer características espaciales. Luego, estas secuencias fueron procesadas por una capa LSTM encargada de aprender los patrones temporales de los movimientos labiales. El modelo fue entrenado en Google Colab Pro con validación cruzada y callbacks para early stopping y checkpoints.

Finalmente, se evaluó el desempeño del modelo con métricas estándar de clasificación multiclase, como la precisión y la matriz de confusión. No se ha implementado aún la etapa de inferencia en tiempo real, ya que esta se encuentra prevista para una fase futura del proyecto.

Este enfoque modular permitió validar paso a paso cada etapa del sistema, facilitando la detección de errores y la optimización de resultados parciales antes de integrarlos en la arquitectura final.

4. Desarrollo del prototipo y resultados

En este capítulo se describe el proceso de desarrollo del prototipo, incluyendo las fases de programación, entrenamiento de modelos y pruebas experimentales. Se exponen los principales resultados obtenidos, se analizan los componentes que integran el sistema y se detallan los retos técnicos que surgieron a lo largo del proyecto, junto con las soluciones adoptadas.

4.1. Recolección y preparación de datos

El primer paso fue diseñar un mecanismo efectivo para generar el dataset. Inicialmente se intentó grabar vídeos manualmente, pero este enfoque resultó ineficiente y propenso a errores. Para resolver este problema, se desarrolló una aplicación web personalizada que guía al usuario durante la pronunciación de los números del 1 al 10. Esta herramienta muestra un contador visual sincronizado con la grabación automática de vídeo desde la cámara, asegurando así un etiquetado preciso y homogéneo.

Para esto, utilice mis conocimientos de técnico superior de desarrollo de aplicaciones web y cree una interfaz grafica mediante JavaScript HTML y CSS capaz de grabar con la cámara indicándote el numero a mostrar. Ademas permite seleccionar cuantas rondas quieres hacer y el numero de veces que vas a decir un número.

Figura 3. Aplicación Web Grabación del dataset.

Con el fin de mantener un entorno de grabación controlado y estructurado, decidí llevar un seguimiento detallado tanto de las personas participantes como de los vídeos generados. Para ello, utilicé dos archivos CSV: uno dedicado a los usuarios y otro al registro de vídeos.

El archivo de usuarios contiene información básica como el identificador (id), nombre, edad y sexo de cada participante.

Tabla 1. Videos

id	id_video	fecha	id_usuario	lugar	Hora
1	7000	25/04/2025	7	escritorio	15:15:00
2	7000	25/04/2025	5	escritorio	11:30:00
3	1000	25/04/2025	5	escritorio	14:30:00
4	4000	25/04/2025	1	comedor	10:30:00
5	5000	28/04/2025	5	escritorio	11:15:00
6	3000	28/04/2025	5	escritorio	11:00:00
7	1000	28/04/2025	1	escritorio	10:15:00
8	4000	28/04/2025	4	comedor	8:45:00
9	2000	28/04/2025	2	escritorio	14:00:00
10	3000	10/05/2025	3	escritorio	9:15:00
11	5000	10/05/2025	5	escritorio	13:30:00
12	7000	10/05/2025	7	escritorio	10:30:00
13	2000	10/05/2025	2	escritorio	9:00:00
14	1000	11/05/2025	1	escritorio	16:00:00
15	5000	14/05/2025	5	escritorio	15:45:00
16	3000	14/05/2025	3	escritorio	11:15:00
17	2000	14/05/2025	2	escritorio	9:45:00
18	4000	14/05/2025	4	comedor	14:15:00

El archivo de vídeos contiene 18 registros, cada uno asociado a un usuario, indicando el identificador del vídeo (id_video), la fecha de grabación, el lugar donde se realizó (escritorio o comedor) y la hora correspondiente. Esto permite llevar un control no solo del volumen de datos, sino también del contexto en que fueron obtenidos.

Tabla 2. Usuarios

Id	nombre	edad	sexo
1	Paula	24	F
2	Andres	53	M
3	Adrián	27	M
4	Sonia	51	F
5	Violeta	20	F
7	Raul	22	M

Distribución de género

En el siguiente gráfico de barras se observa la distribución de participantes por género. El dataset está perfectamente balanceado en cuanto a género, lo cual es positivo para evitar sesgos relacionados con características faciales específicas.

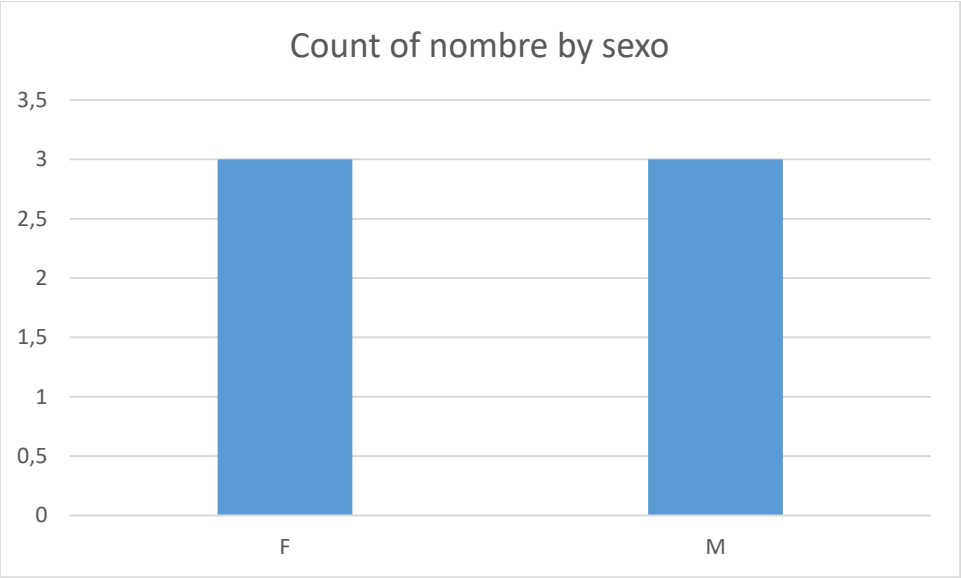


Figura 4. Gráfico distribución de género

Distribución por lugar de grabación

En cuanto al lugar de grabación, la mayoría de las sesiones se realizaron en un entorno controlado frente a un escritorio. Sin embargo, existe un único caso de grabaciones en el comedor, correspondiente al usuario Violeta (ID 5). Este desbalance podría ser relevante en términos de iluminación, fondo y posibles variaciones en la detección de landmarks faciales.

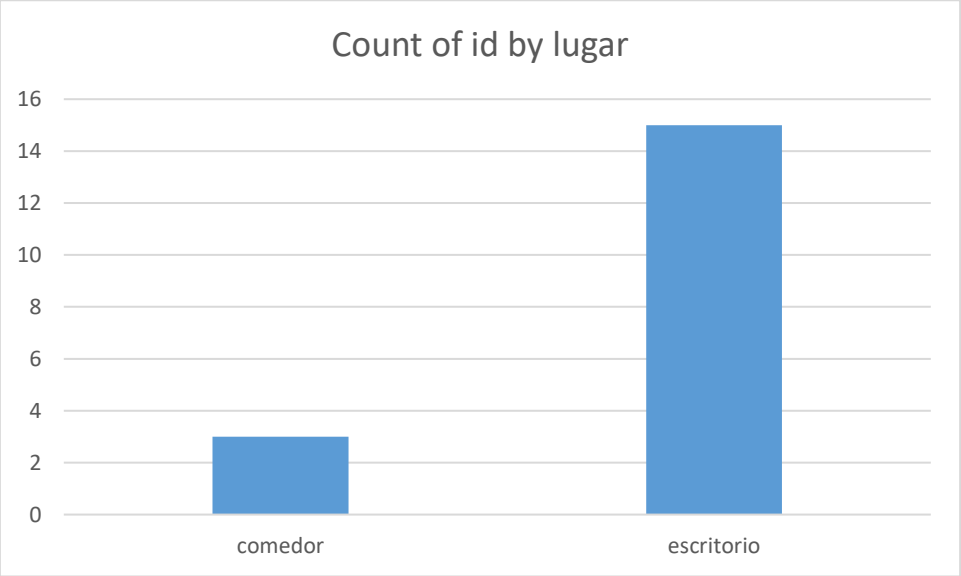


Figura 5. Gráfico Distribución por lugar de grabación

Gestión del trabajo y organización

Para planificar y controlar el avance del proyecto, he utilizado la herramienta ClickUp, un sistema de gestión de tareas que me ha permitido definir subtareas, asignar tiempos estimados y establecer prioridades en cada fase del desarrollo. Gracias a esta organización he podido afrontar mejor la carga de trabajo, dividir el proyecto en entregas manejables y mantener una visión clara del progreso, incluso teniendo un tiempo diario muy limitado.

Esta herramienta ha sido especialmente útil para no perder de vista tareas críticas como la grabación controlada, el aumento de datos, la integración de MediaPipe o el ajuste de hiperparámetros en el modelo. El seguimiento visual del avance me ha ayudado también a evitar bloqueos y reducir el estrés en momentos de carga académica elevada.

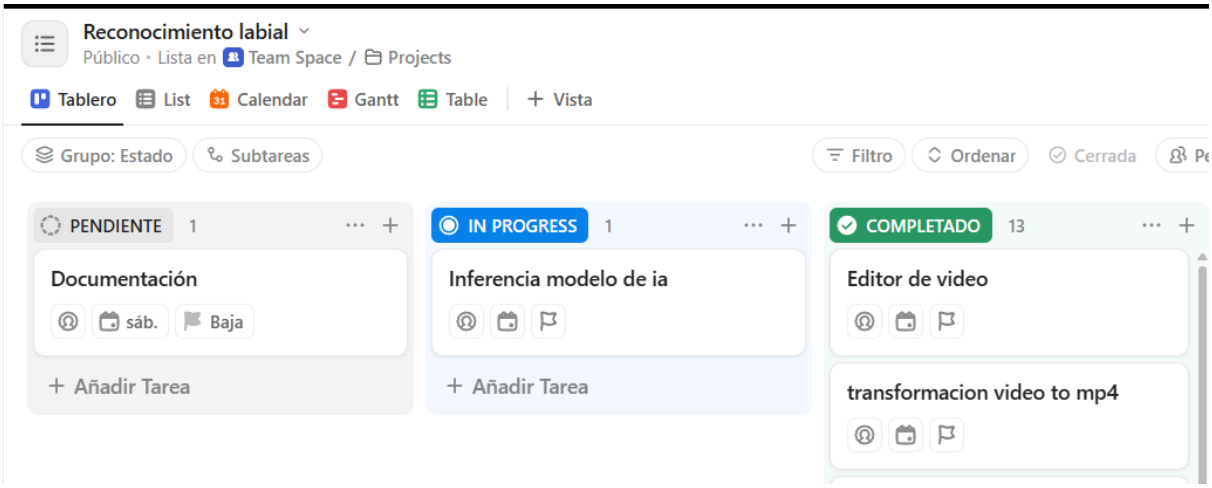


Figura 6. Imagen de ClickUp en mi proyecto de reconocimiento Labial.

Cada grabación se almacena en formato .webm y posteriormente se convierte a .mp4 mediante FFMPEG. Los vídeos convertidos se segmentan en fragmentos de 2 segundos y se organizan en carpetas numeradas, una por clase, formando así la base del dataset.

A continuación, se aplicaron técnicas de Data Augmentation con Albumentations. Se generaron cinco versiones aumentadas por vídeo, con transformaciones visuales como rotación, inversión horizontal, desenfoque, cambios de brillo y compresión. Esto permitió incrementar la diversidad del conjunto de datos sin necesidad de realizar nuevas grabaciones.

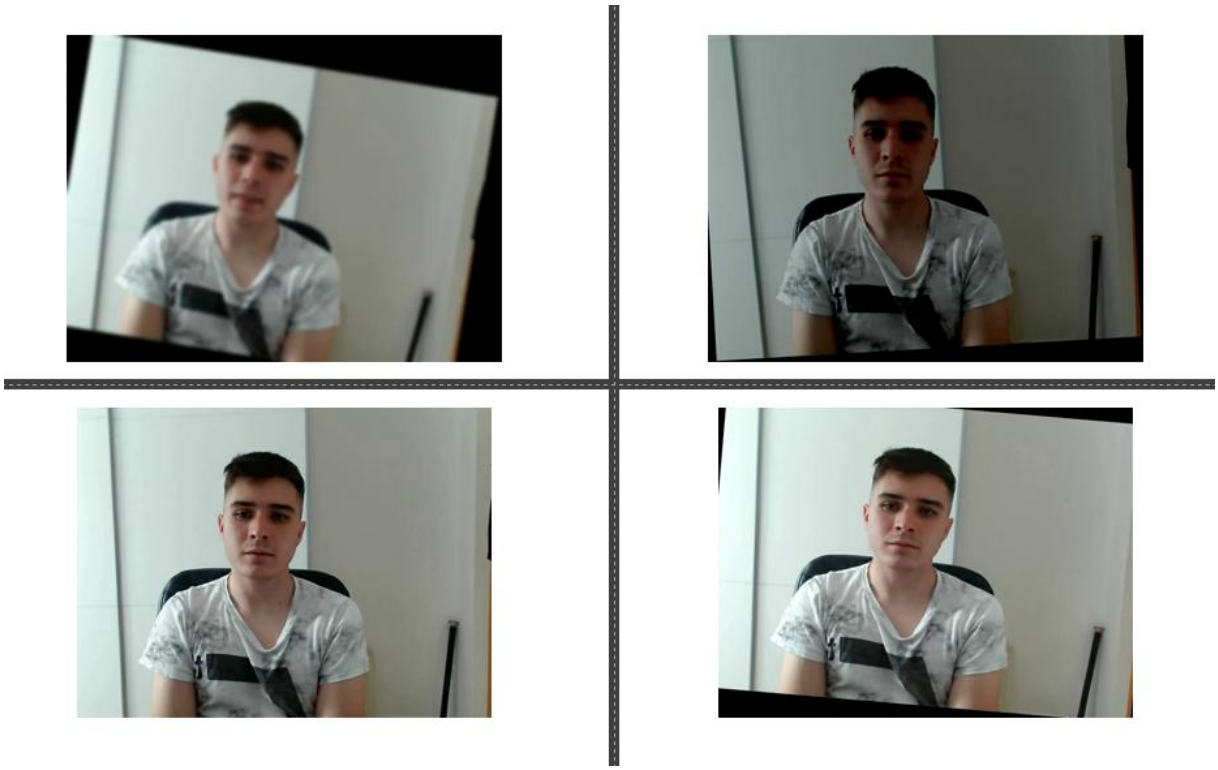


Figura 7. Ejemplos de data aumentación con pequeñas rotaciones en las imágenes y cambios de brillo y de nitidez.

Después del aumento, los vídeos fueron convertidos en secuencias de imágenes (frames). Cada secuencia consta de 30 fotogramas ordenados cronológicamente. Luego, se aplicó MediaPipe Face Mesh para detectar los puntos faciales clave y recortar dinámicamente la región de los labios. Las imágenes recortadas se normalizaron y se almacenaron en escala de grises como arrays NumPy de tamaño uniforme.

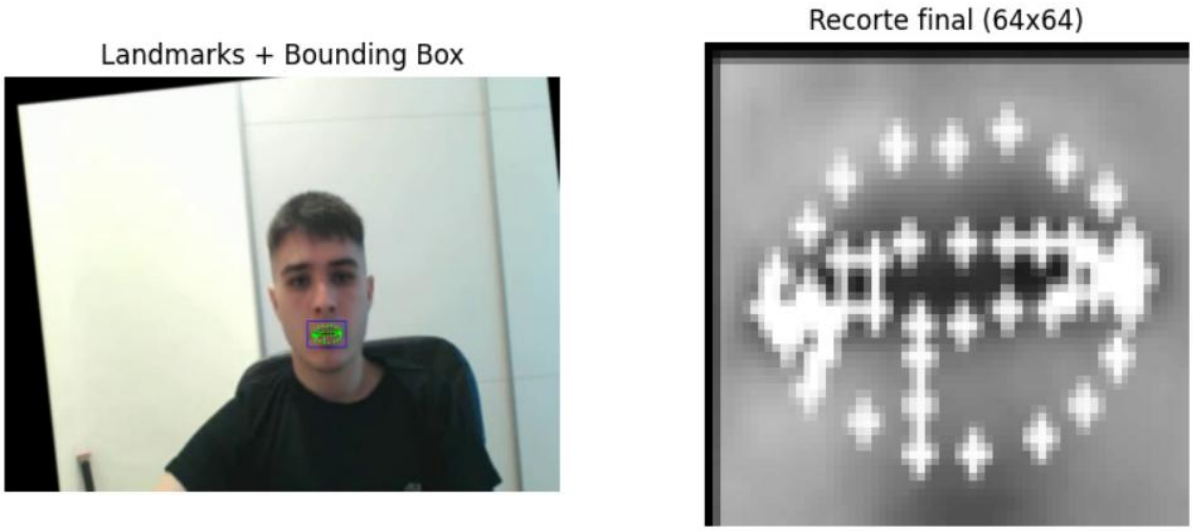


Figura 8. Ilustración representativa del rostro humano con los puntos específicos que MediaPipe detecta en la región de la boca.

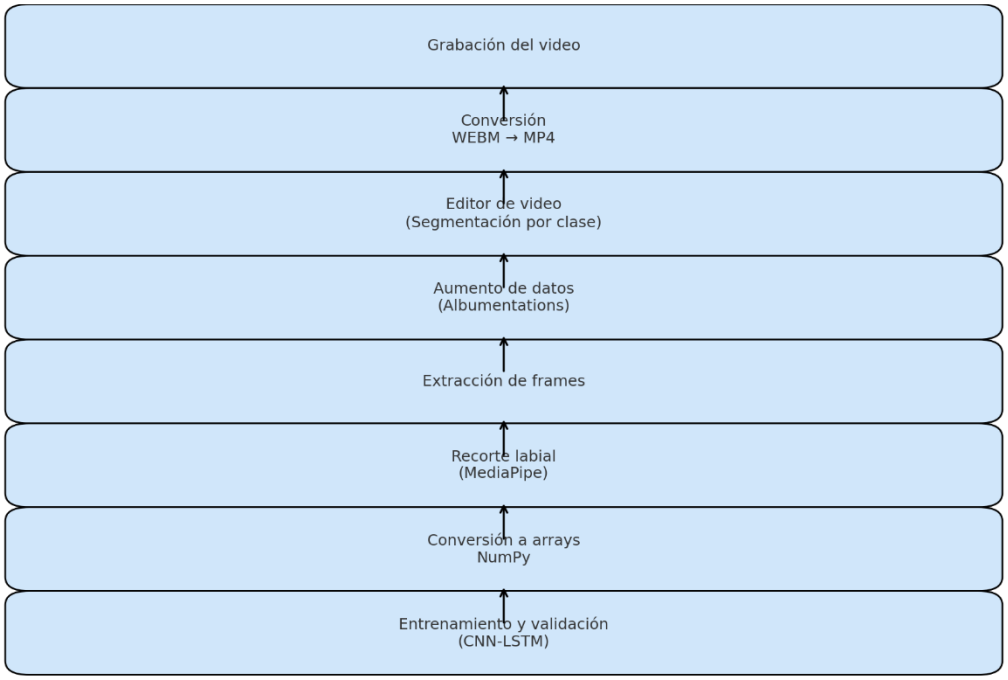


Figura 9. Diagrama de flujo desde la grabación hasta el modelo

4.2 Arquitectura del modelo

El sistema de reconocimiento de palabras mediante lectura de labios ha sido implementado mediante un modelo de redes neuronales profundas que combina dos tipos de estructuras: redes convolucionales (CNN) para la extracción de características espaciales de cada fotograma, y redes neuronales recurrentes del tipo LSTM para el análisis temporal de las secuencias. Esta arquitectura híbrida CNN-LSTM ha demostrado ser especialmente eficaz en tareas que requieren entender cómo evoluciona una imagen en el tiempo, como ocurre al pronunciar una palabra con los labios.

Entrada del modelo

El modelo recibe como entrada una secuencia de vídeo ya procesada, compuesta por 30 imágenes en escala de grises de 128 × 128 píxeles, que representan los movimientos labiales de una persona al pronunciar un número del 1 al 10. Esta secuencia se presenta como un tensor de forma (30, 128, 128, 1), 30 es el número de frames, 128×128 es la dimensión espacial de cada imagen, 1 representa el canal (escala de grises).

Estas imágenes se han obtenido tras aplicar técnicas de recorte dinámico de labios con MediaPipe y normalización.

Extracción de características espaciales – CNN (TimeDistributed)

Cada fotograma de la secuencia se procesa individualmente a través de capas convolucionales (Conv2D), que tienen la función de identificar patrones visuales relevantes, como bordes, líneas, formas, contornos y texturas que describen la forma del labio en cada instante.

Para poder aplicar la misma red convolucional a cada imagen sin perder la dimensión temporal de la secuencia, se utiliza el envoltorio TimeDistributed, una técnica que aplica las mismas operaciones de forma independiente a cada elemento de una secuencia temporal. Esto permite que las capas convolucionales operen sobre cada frame como si fueran imágenes aisladas, conservando al mismo tiempo el orden de aparición.

La estructura emplea dos capas Conv2D, con filtros de diferentes profundidades. La primera tiene 16 filtros y la segunda 32, ambas con kernels de tamaño 3x3. Entre ambas se inserta una capa MaxPooling2D, que reduce la resolución espacial de los mapas de activación, conservando solo las características más importantes y reduciendo el tamaño del tensor de salida. Esta operación ayuda también a disminuir el sobreajuste y mejora la eficiencia computacional.

Tras pasar por estas capas, cada fotograma queda convertido en un mapa de características reducido, que encapsula la información más significativa de la forma labial para ese instante temporal.

Representación vectorial – Flatten

Después de las capas convolucionales, el resultado de cada imagen se aplana mediante la capa Flatten, transformando el mapa 2D de características en un vector unidimensional. Gracias al envoltorio TimeDistributed, el resultado final es una secuencia de 30 vectores que representan las características extraídas de los 30 fotogramas originales.

Esta secuencia de vectores sirve como entrada para la red LSTM, que se encargará de aprender la evolución temporal de estos patrones a lo largo del tiempo.

Análisis temporal – LSTM

La capa LSTM (Long Short-Term Memory) juega un papel fundamental en la arquitectura del modelo, ya que es la responsable de identificar cómo cambian las características labiales a medida que avanza el tiempo. Las redes LSTM están diseñadas para manejar secuencias de datos y aprender relaciones a largo plazo entre elementos de la secuencia.

En lugar de procesar cada vector de forma aislada, la LSTM analiza cómo los patrones se transforman desde el primer fotograma hasta el último. Esto es esencial en la lectura de labios, ya que muchas palabras pueden tener formas labiales parecidas en ciertos instantes, pero la dinámica global del movimiento permite diferenciarlas correctamente.

La arquitectura de una LSTM incluye tres puertas: una de entrada, una de olvido y una de salida. Estas puertas permiten controlar el flujo de información a través del tiempo, decidiendo qué datos se conservan, cuáles se olvidan y cómo se genera la salida en cada paso de la secuencia. Gracias a este mecanismo, la LSTM es capaz de mantener en su "memoria" información relevante que puede aparecer o ser necesaria varios fotogramas después.

Capa Dropout

Después de la LSTM, se incluye una capa Dropout, que actúa como regularizador del modelo. Durante el entrenamiento, esta capa "apaga" aleatoriamente un porcentaje de neuronas, lo que obliga a la red a no depender demasiado de un subconjunto concreto de nodos. Esta técnica es clave para reducir el sobreajuste, especialmente en modelos con gran capacidad como las redes neuronales profundas.

Clasificación final – Capa densa (softmax)

La última etapa del modelo es una capa completamente conectada (Dense) con activación softmax, que transforma la salida de la LSTM en una distribución de probabilidad sobre las 10 clases posibles (los números del 1 al 10). La clase con mayor probabilidad se considera la predicción final del sistema.

Entrenamiento del modelo

El proceso de entrenamiento del modelo se llevó a cabo utilizando el entorno colaborativo de Google Colab Pro, aprovechando sus capacidades de cómputo avanzadas, en particular la disponibilidad de unidades de procesamiento gráfico (GPU). Esta infraestructura fue fundamental para acelerar el entrenamiento, dado que el modelo CNN-LSTM utilizado maneja secuencias de imágenes y posee una alta demanda computacional.

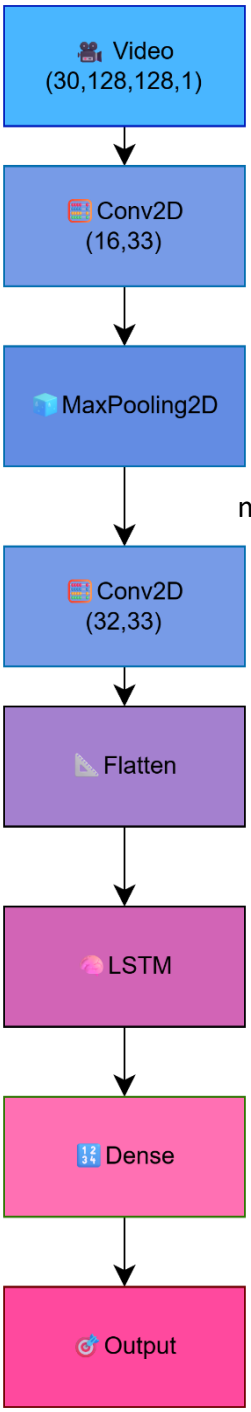
Para la optimización del aprendizaje, se utilizó el algoritmo Adam, un método adaptativo ampliamente reconocido por su eficiencia y estabilidad al entrenar redes neuronales profundas. Este optimizador ajusta automáticamente las tasas de aprendizaje para cada parámetro, permitiendo converger más rápidamente hacia un mínimo de la función de pérdida. Dicha función de pérdida fue definida como la entropía cruzada categórica, la cual resulta especialmente adecuada en contextos de clasificación multiclase, como es el caso del presente proyecto, en el que se desea asignar cada secuencia a una de diez clases posibles.

La métrica principal seleccionada para evaluar el rendimiento del modelo durante el entrenamiento fue la precisión, o accuracy, medida sobre un conjunto de validación

independiente. Esta métrica permite observar de forma clara la proporción de predicciones correctas realizadas por el modelo en datos no vistos previamente.

Adicionalmente, se implementaron mecanismos de control automático para garantizar un entrenamiento eficiente y evitar el sobreajuste. Uno de ellos fue el early stopping, el cual detiene el proceso de entrenamiento en cuanto la métrica de validación deja de mejorar durante un número determinado de épocas, evitando así que el modelo continúe ajustándose a detalles irrelevantes del conjunto de entrenamiento. Junto a esto, se empleó un sistema de model checkpoint, encargado de guardar automáticamente los pesos de la red en el punto donde se obtuvo el mejor rendimiento sobre la validación, permitiendo así recuperar el modelo más eficaz alcanzado durante todo el proceso.

El modelo fue alimentado por lotes mediante un generador personalizado que organizaba el conjunto de datos de forma eficiente. Este generador no solo se encargaba de dividir las muestras en bloques manejables durante la fase de entrenamiento, sino que además normalizaba las imágenes de entrada y transformaba las etiquetas de clase en vectores de codificación one-hot, lo cual es esencial para que la capa de salida softmax pueda interpretarlas correctamente durante el aprendizaje supervisado.



Una secuencia de 30 imágenes (frames) en escala de grises, cada una de tamaño 128×128 píxeles.

Aquí se aplica la primera capa convolucional a cada imagen. Se utilizan 16 filtros con un tamaño de 3×3 píxeles. Esta capa detecta patrones visuales básicos como bordes, líneas y curvas en los labios, que son fundamentales para capturar la forma labial en cada instante.

reduce el tamaño de los mapas de características seleccionando solo los valores más importantes en regiones específicas. Esto hace que el modelo sea más eficiente y resistente al ruido visual,

Una segunda capa convolucional extrae características más complejas, esta vez con 32 filtros.

Cada mapa de características generado por las capas anteriores es aplanado (convertido en un vector unidimensional).

Esta capa analiza la evolución temporal de los vectores anteriores. Gracias a su memoria interna, la red LSTM puede identificar cómo cambian los labios a lo largo de los 30 fotograma

La salida de la LSTM se conecta a una capa completamente conectada (o "densa"), que resume toda la información temporal en un vector único que representa la probabilidad de pertenencia a cada clase.

Finalmente, la salida pasa por una función de activación softmax, que convierte los valores anteriores en una distribución de probabilidad

Figura 10. Esquema CNN-LSTM con bloques:
TimeDistributed CNN → LSTM → Dense

4.3 Evaluación del sistema

La evaluación del sistema se ha estructurado en dos partes: por un lado, el análisis del rendimiento general del modelo CNN-LSTM entrenado sobre el dataset y por otro, una prueba específica de inferencia real utilizando nuevos datos grabados y procesados por el autor.

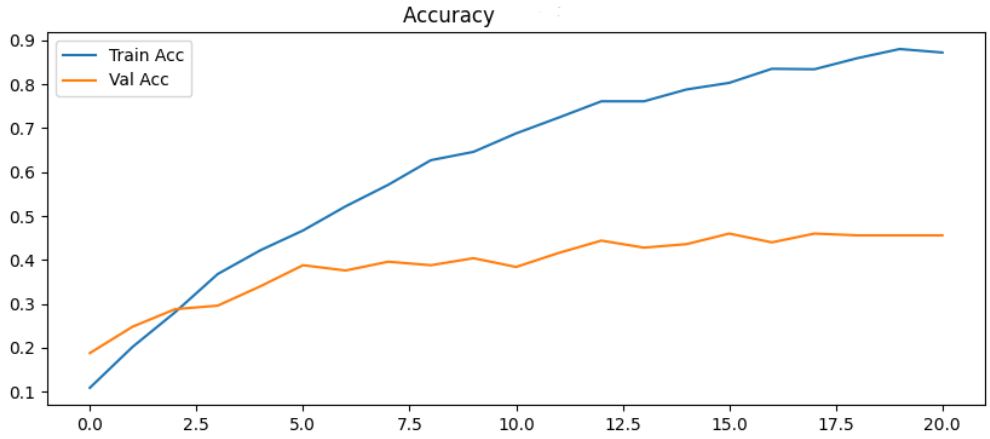


Figura 11. Accuracy Por Epoca

Se observa que el modelo alcanza una precisión del 88% sobre el conjunto de entrenamiento, pero su precisión sobre el conjunto de validación se estabiliza en torno al 46%, indicando un posible sobreajuste.

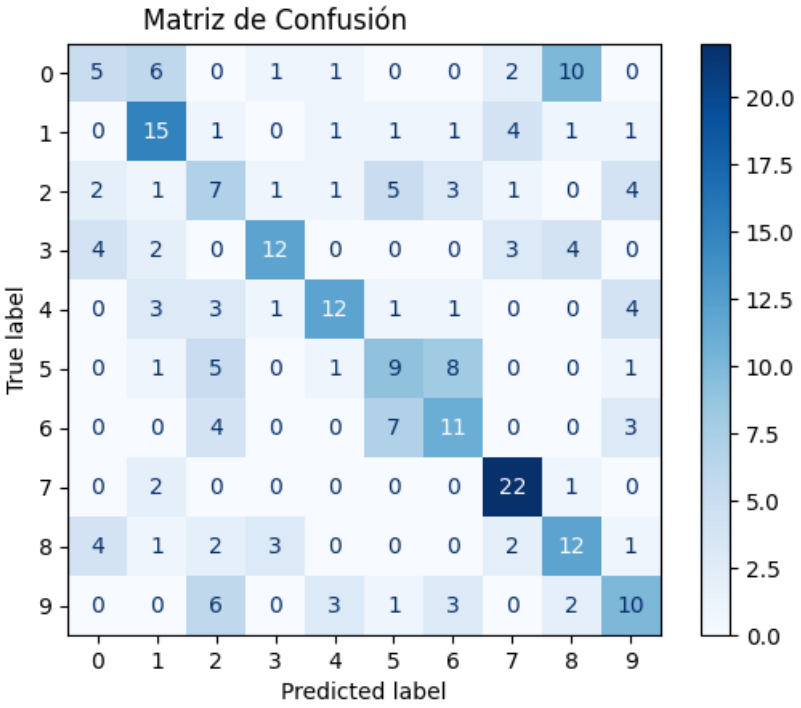


Figura 12. Matriz de confusion

El modelo tiene un desempeño variable entre clases. Por ejemplo:

La clase **7** (número ocho) se predice con alta precisión (22 de 25 aciertos).

Las clases 0,2, tienden a tener error y la clases 8 la confunde mucho con la clase 0.

Tabla 3. Métricas por clase

Clase	Precisión	Recall	F1-score
0	0.33	0.20	0.25
1	0.48	0.60	0.54
2	0.25	0.28	0.26
3	0.67	0.48	0.56
4	0.63	0.48	0.55
5	0.38	0.36	0.37
6	0.41	0.44	0.42
7	0.65	0.88	0.75
8	0.40	0.48	0.44
9	0.42	0.40	0.41

Tabla 4.Promedios

Info	Resultado
Precisión global	46%
F1-score macro	45.34%
Clase más robusta	“7”
Clase más débil	“0” y “2”

Estos resultados reflejan un modelo funcional, pero con margen de mejora. La diferencia entre entrenamiento y validación sugiere que el modelo podría beneficiarse de más regularización, más diversidad de datos y un mayor número de usuarios en el dataset.

Prueba de inferencia en entorno real

Para simular un caso de uso práctico, se grabó un nuevo video con pronunciaciones y se procesó siguiendo todo el procedimiento: segmentación, aumento, extracción de frames, detección de labios y conversión a arrays .npy.

Tabla 5. Inferencia

Archivo	Real	Predicha	Confianza
inferencia_10_0009.npy	10	10	0.50
inferencia_10_0019.npy	10	10	0.48
inferencia_1_0010.npy	1	1	0.44
inferencia_3_0012.npy	3	1	0.43
inferencia_3_0002.npy	3	10	0.47

Acertó 3 de 5 muestras y una precisión final del 60%.

Durante esta fase también se detectó que el modelo no era capaz de distinguir correctamente ciertas muestras, incluso tras el preprocesamiento. Estas muestras, con detecciones de labios erráticas o altamente ambiguas, fueron descartadas de la evaluación para no distorsionar los resultados.

Además, se observaron **confusiones frecuentes en la clase 3**, consistente con los errores detectados en la matriz de confusión general del modelo. Esto evidencia que hay clases que, por sus movimientos labiales similares, generan ambigüedad para el sistema actual.

Estas confusiones han sido producidas ya que el entorno de prueba de inferencia no es el mismo que el de grabación para no crear un entorno idéntico.

4.4 Problemas técnicos encontrados y soluciones aplicadas

Durante el desarrollo del proyecto surgieron múltiples desafíos técnicos y logísticos que requirieron no solo creatividad, sino también una gran capacidad de adaptación para poder cumplir los objetivos en los plazos establecidos. Uno de los primeros problemas fue la limitada capacidad de almacenamiento del equipo de trabajo, el cual dispone de un disco duro de únicamente 500 GB. A medida que avanzaban las fases del proyecto —especialmente tras la

generación del conjunto de datos y la aplicación de técnicas de aumento con Albumentations— el volumen de información creció de forma considerable. Entre vídeos originales, aumentados, secuencias de frames, arrays procesados y resultados de prueba, el total superó los 100 GB, saturando rápidamente el espacio disponible. Este obstáculo fue superado temporalmente mediante una constante gestión manual de los archivos, aunque también condicionó la posibilidad de conservar versiones intermedias y modelos descartados.

Otro aspecto importante fue la necesidad de contar con recursos computacionales de alto rendimiento para poder entrenar modelos con datos visuales en secuencia, algo que excedía las capacidades de procesamiento del equipo personal. Ante esta situación, se optó por utilizar Google Colab Pro, un entorno de desarrollo colaborativo en la nube que ofrece acceso a GPU de gama alta. Esta decisión implicó una inversión económica adicional, ya que el servicio requiere una suscripción mensual de 11.99 € (precio actual), coste que asumí personalmente como parte del compromiso con el desarrollo del sistema.

Además de los desafíos técnicos, surgieron problemas relacionados con la gestión de entornos de trabajo en Python. Debido a incompatibilidades entre ciertas versiones de librerías críticas, como OpenCV, MediaPipe, TensorFlow o Albumentations, fue necesario mantener dos entornos separados en Anaconda, cada uno configurado de forma específica para tareas distintas. Esta fragmentación dificultó la integración y prueba continua del sistema, obligándome a cambiar entre entornos en función de cada fase del proyecto, lo cual ralentizó el proceso de desarrollo.

A estas dificultades técnicas se sumó una limitación de tiempo considerable. El proyecto fue desarrollado en un plazo inferior a un mes, con una disponibilidad media de solo 4 horas diarias. Este tiempo debía repartirse, además, con la realización de otro trabajo final obligatorio del curso, lo que supuso una carga académica excepcionalmente alta. El compromiso con ambos proyectos, junto con el esfuerzo requerido para superar las barreras técnicas mencionadas, implicó un nivel elevado de dedicación personal y gestión del estrés.

A pesar de estas dificultades, cada reto fue abordado con una actitud proactiva y resolutiva, y finalmente se logró alcanzar un prototipo funcional sólido, capaz de reconocer palabras mediante lectura labial automatizada. Las decisiones tomadas —como el uso de Colab Pro, la segmentación de entornos o la optimización del almacenamiento— fueron clave para continuar avanzando sin comprometer la calidad del sistema desarrollado.

5. Conclusiones y trabajo futuro

Las conclusiones del trabajo permiten valorar los objetivos alcanzados y las principales lecciones aprendidas. Asimismo, se proponen líneas de mejora para el futuro, considerando la ampliación del vocabulario reconocido y la mejora de la robustez del sistema.

A lo largo del proceso he podido experimentar de primera mano lo complejo que puede resultar llevar un modelo de inteligencia artificial desde una idea hasta una implementación realista, capaz de ejecutar inferencias sobre nuevos datos. Pero lo más importante es que este proyecto no ha sido simplemente una tarea académica más: ha sido una especie de laboratorio personal, donde he podido combinar todos los conocimientos adquiridos en el curso con herramientas reales, datos reales, y limitaciones reales.

5.1.Conclusiones del trabajo

Este proyecto ha supuesto una experiencia de aprendizaje profundamente enriquecedora, no solo desde el punto de vista técnico, sino también personal. Lo que inicialmente era una idea ambiciosa —construir un sistema de reconocimiento de palabras a partir del movimiento de los labios— ha terminado convirtiéndose en un conjunto funcional de herramientas y modelos que permiten lograrlo, aunque aún con limitaciones.

Uno de los logros más importantes ha sido, sin duda, el diseño completo de todas las fases necesarias para abordar el problema de lectura labial. Aunque no se ha tratado de un pipeline automatizado en sentido estricto, sí he desarrollado manualmente, con scripts independientes y bien definidos, cada una de las etapas del proceso: desde la grabación de vídeos sincronizada con un contador, pasando por su conversión a formato compatible, la segmentación temporal, el aumento de datos con técnicas de visión artificial, la extracción de frames, la localización de la región labial con MediaPipe, y finalmente la preparación del input para la red neuronal. Poder recorrer todo ese camino, controlando cada fase y observando cómo los datos evolucionan desde un vídeo en crudo hasta una predicción de clase, ha sido muy revelador.

Durante el entrenamiento del modelo CNN-LSTM, se alcanzó una precisión del 88% sobre el conjunto de entrenamiento, lo que indica que el sistema ha sido capaz de aprender patrones visuales y temporales complejos. Sin embargo, al aplicar el modelo a datos de validación, la precisión descendía al 46%, lo cual deja ver un problema clásico de sobreajuste: el modelo se adapta bien a los datos que ha visto, pero tiene dificultades para generalizar. Aun así, estos resultados siguen siendo prometedores, especialmente si se tiene en cuenta que se ha entrenado con un dataset propio, limitado en número de participantes y variedad.

Lo más interesante fue comprobar su funcionamiento ante una prueba de inferencia real: cinco muestras nunca vistas, procesadas con el mismo flujo de trabajo. En esa prueba, el modelo logró predecir correctamente 3 de los 5 casos. Aunque la muestra es reducida, este pequeño experimento confirmó que el sistema tiene una base funcional. Cabe mencionar que algunas muestras tuvieron que descartarse porque la detección de labios fallaba o el modelo no era capaz de diferenciar correctamente entre clases. Este tipo de problemas no son fallos del todo inesperados: revelan precisamente las debilidades del sistema ante condiciones no ideales, como variaciones en la postura, en la pronunciación o en la iluminación.

Una observación importante es que ciertas clases, como el número 3, han generado sistemáticamente errores de predicción. Este fenómeno se observa tanto en la matriz de confusión del modelo como en las pruebas reales. La conclusión evidente es que hay números cuya pronunciación produce movimientos labiales demasiado similares entre sí, y que eso complica mucho su clasificación solo con información visual. Este tipo de dificultades, lejos de ser frustrantes, me han servido para valorar con más realismo los límites del enfoque basado únicamente en vídeo sin audio.

En conjunto, este trabajo me ha permitido aplicar en la práctica una gran cantidad de conocimientos teóricos: desde visión por computador hasta redes neuronales, pasando por ingeniería de datos y evaluación de modelos. Pero quizás lo más valioso ha sido desarrollar una conciencia crítica sobre el desarrollo de sistemas inteligentes: no basta con que un modelo “funcione” en entrenamiento, es necesario garantizar que sea útil, estable y robusto en

situaciones reales. Ese será, sin duda, uno de mis principales focos en futuras extensiones del proyecto.

5.2. Líneas de trabajo futuro

Pensar en el futuro de este proyecto es casi inevitable, sobre todo porque, al terminar esta etapa, no se siente como un punto final, sino más bien como el cierre de una primera fase de exploración. Lo que tengo entre manos no es un producto terminado, sino una base sobre la cual se pueden construir cosas mucho más potentes.

Una de las cosas que más he notado es la necesidad de que el sistema cuente con más diversidad en su conjunto de datos. Aunque el modelo ha sido entrenado con muestras variadas, el número de participantes sigue siendo limitado. Esto provoca que el modelo, sin quererlo, se habitúe a ciertos gestos o formas de pronunciar que no generalizan bien a otros hablantes. Si realmente quiero que este sistema sea capaz de funcionar con cualquier persona, entonces el dataset tiene que incorporar muchas más voces, rostros y estilos de pronunciación. Idealmente, debería poder aprender no solo de las formas, sino también de las variaciones entre hablantes.

También considero clave explorar otras arquitecturas. Aunque la CNN-LSTM ha demostrado ser funcional, existen modelos más avanzados que están diseñados específicamente para entender secuencias visuales con mayor profundidad. Por ejemplo, el uso de redes bidireccionales (BiLSTM) o la implementación de transformers visuales podría aportar una capacidad adicional para captar diferencias sutiles entre palabras. Además, una mejora que me parece especialmente prometedora es la integración del flujo óptico (optical flow), que permitiría al sistema detectar con mayor precisión cómo se mueven los labios entre frame y frame, captando detalles que incluso una red convolucional podría pasar por alto.

Y finalmente, no puedo evitar imaginar el sistema funcionando en tiempo real. Aunque esto aún no se ha implementado, ya hay una buena parte del trabajo hecha: la normalización, la entrada de frames, la predicción. Faltaría solo conectar esas piezas en una interfaz capaz de capturar desde cámara, ejecutar la predicción y mostrar el resultado al usuario. Puede parecer un detalle, pero tener un sistema que responda en vivo a lo que una persona dice sin emitir sonido, abre una puerta enorme a aplicaciones inclusivas, especialmente para personas con discapacidades auditivas, o para entornos donde el ruido impide una comunicación clara.

Este proyecto me ha enseñado que construir modelos es solo una parte del proceso. La verdadera dificultad está en integrar esos modelos con datos reales, bajo condiciones reales, y aun así obtener resultados que sean útiles. Eso es lo que he intentado hacer aquí, y eso es lo que me motiva a seguir explorando.

Referencias bibliográficas

Deocampo, N. J., Villarica, M., & Vinluan, A. (2024). A Lip-Reading Model for Tagalog Using Multimodal Deep Learning Approach. International Journal of Computing Sciences Research, 8, 2796-2808. https://doi.org/ 10.25147/ijcsr.2017.001.1.186

Abbasi, A. F., Yousefi-Koma, A., Firouzabadi, S. D., Rashidi, P., & Naeini, A. (2025). Integrating Persian lip reading in Surena-V humanoid robot for human-robot interaction. arXiv. https://arxiv.org/abs/2501.13996