UNIVERSIDAD DE VALLADOLID

ESCUELA TÉCNICA SUPERIOR DE INGENIEROS DE TELECOMUNICACIÓN

TRABAJO FIN DE MÁSTER

MÁSTER UNIVERSITARIO EN INVESTIGACIÓN EN TECNOLOGÍAS DE LA INFORMACIÓN Y LAS COMUNICACIONES

HACIA UN SISTEMA DE RECONOCIMIENTO DEL LENGUAJE CORPORAL EN PRESENTACIONES ORALES UTILIZANDO TÉCNICAS DE *MACHINE LEARNING*

Autora:

Yulieth Hoyos Vera

Tutor:

Ioannis Dimitriadis Damouils

Valladolid, 20 de septiembre de 2024

TÍTULO: Hacia un sistema de reconocimiento del lenguaje corporal en presentaciones orales utilizando técnicas de *machine learning*

AUTORA: **Yulieth Hoyos Vera**

TUTOR: **Ioannis Dimitriadis Damoulis**

DEPARTAMENTO: **Departamento de Teoría de la Señal y Comunicaciones e Ingeniería Telemática**

TRIBUNAL

PRESIDENTE: **D. Ignacio de Miguel Jiménez**

VOCAL: **D. Mario Martínez Zarzuela**

SECRETARIO: **D. Santiago Aja Fernández**

SUPLENTE: **Dña. Miriam Antón Rodríguez**

SUPLENTE: **D. Jesús Poza Crespo**

SUPLENTE: **D. José Ramón Durán Barroso**

FECHA: **20 de septiembre de 2024**

CALIFICACIÓN:

Resumen

Las instituciones educativas enfrentan el desafío de evaluar y mejorar las competencias del siglo 21 en los estudiantes. Estas competencias son fundamentales para su desarrollo académico y profesional. La comunicación no verbal, que incluye gestos y posturas, es especialmente importante para enriquecer la competencia de comunicación oral. Esta habilidad permite mostrar emociones y actitudes que respaldan una presentación oral efectiva. Para complementar la evaluación integral de estas competencias, se propone utilizar un sistema de evaluación objetiva del lenguaje corporal. Este sistema utiliza algoritmos de *machine learning* y visión por computadora para analizar el lenguaje corporal de los estudiantes y proporciona datos cuantitativos que identifican áreas específicas de mejora para cada estudiante. A diferencia de las evaluaciones manuales, que son subjetivas y sesgadas, este sistema automatizado puede ahorrar tiempo y mejorar la eficacia de la evaluación en entornos académicos y profesionales. El sistema de evaluación automática del lenguaje corporal puede brindar a los profesores una visión cuantificable de los gestos positivos y negativos durante las presentaciones de los estudiantes. Además, brinda a los alumnos la oportunidad de ver cómo obtuvieron esos resultados de manera consistente a través del sistema. Esto les permite mejorar su comunicación efectiva durante las presentaciones. El sistema desarrollado tiene el potencial de mejorar significativamente la efectividad de la comunicación en diversos contextos, desde la educación hasta los entornos empresariales y políticos. Al proporcionar información útil sobre el lenguaje corporal de los presentadores, el sistema propuesto puede contribuir a un mejor desempeño en estas áreas.

Palabras clave

Visión por computadora, habilidades comunicativas, *machine learning*

Abstract

Educational institutions face the challenge of assessing and improving 21st century competencies in students. These competencies are critical to their academic and professional development. Nonverbal communication, which includes gestures and postures, is especially important to enrich oral communication competence. This skill allows for the display of emotions and attitudes that support an effective oral presentation. To complement the comprehensive assessment of these competencies, it is proposed to use an objective body language assessment system. This system uses machine learning and computer vision algorithms to analyze students' body language and provides quantitative data that identifies specific areas of improvement for each student. Unlike manual assessments, which are subjective and biased, this automated system can save time and improve the effectiveness of assessment in academic and professional settings. The automatic body language assessment system can give teachers quantifiable insight into positive and negative gestures during students' presentations. In addition, it gives students the opportunity to see how they consistently achieved those results across the system. This allows them to improve their effective communication during presentations. The developed system has the potential to significantly improve the effectiveness of communication in various contexts, from education to business and political settings. By providing useful information about the body language of presenters, the proposed system can contribute to better performance in these areas.

Keywords:

computer vision, communication skills, machine learning

Agradecimientos

Me gustaría agradecer a mi tutor Ioannis por toda la paciencia y dedicación que me ha brindado a lo largo del desarrollo de este trabajo, y a mi familia por siempre apoyarme en cada paso que doy.

Índice

1. Introducción

La comunicación es una parte fundamental de la interacción humana, y en un mundo cada vez más impulsado por las Tecnologías de Información y las Comunicaciones (TIC), las presentaciones orales desempeñan un papel crucial en la transmisión efectiva de información, ideas y emociones. En este contexto, la comunicación no verbal, que incluye gestos y posturas, ha demostrado ser un componente esencial para comprender y enriquecer la comunicación oral (Holland et al., 2017). La comunicación no verbal a menudo revela emociones, intenciones y actitudes que complementan el discurso verbal (Carney et al., 2010).

La comunicación no verbal, como aspecto fundamental de la comunicación oral, cobra una importancia destacada en contextos académicos, donde la transmisión de información de manera clara y convincente es esencial. Sin embargo, la pandemia de COVID-19 planteó un gran desafío para la comunicación no verbal, dado que el uso generalizado de mascarillas dificultó la expresión facial y la percepción de las emociones. Por consiguiente, el desarrollo de estrategias que fortalezcan el lenguaje corporal en presentaciones orales, reforzando elementos como la postura, los gestos y el contacto visual cobró una importancia aun mayor (Castro, 2023).

La implementación de un sistema automático para la evaluación de presentaciones orales es crucial, ya que ofrece la posibilidad de analizar la comunicación no verbal de manera rápida, eficiente, objetiva y detallada. Utilizando avanzadas técnicas de aprendizaje automático (*machine learning*), este sistema puede identificar y valorar con precisión los elementos del lenguaje corporal, proporcionando así una retroalimentación valiosa para mejorar las habilidades de presentación.

Así pues, contar con un sistema que permita una evaluación más objetiva y precisa del desempeño de los presentadores, ayudaría a complementar la evaluación subjetiva de los presentadores con datos cuantitativos sobre el lenguaje corporal, lo que puede ayudar a identificar áreas de mejora específicas y proporcionar retroalimentación concreta para el desarrollo de habilidades de presentación (D'Mello & Graesser, 2010).

En este sentido, el desarrollo de un sistema basado en *MediaPipe* que es una biblioteca de código abierto desarrollada por Google que facilita el reconocimiento de la posición de las manos, la cara y otras partes del cuerpo humano en imágenes y videos, sería de gran ayuda e interés para este estudio. Este código abierto se utiliza principalmente en aplicaciones de visión por computadora para detectar y rastrear posturas humanas en tiempo real. *MediaPipe* utiliza modelos de aprendizaje automático para identificar puntos clave del cuerpo *(landmarks)* y así analizar patrones de lenguaje corporal, lo que es particularmente útil en tareas como la clasificación de gestos y la evaluación de presentaciones orales (Lugaresi et al., 2019).

Sin embargo, hasta el momento, la mayoría de las herramientas de evaluación se centran en aspectos verbales o en técnicas tradicionales de evaluación manual, como escalas de puntaje o rúbricas. Estas soluciones no son suficientes debido a varias limitaciones: en primer lugar, las herramientas de evaluación manual pueden ser subjetivas y tener sesgos, ya que dependen de la percepción individual de los evaluadores, lo que puede llevar a inconsistencias en la evaluación y a resultados poco confiables (Chen et al., 2014). Por otro lado, evaluación manual del lenguaje corporal en presentaciones orales puede resultar excesivamente laboriosa y consumir un tiempo valioso, lo que a menudo la convierte en una opción poco viable para entornos académicos y profesionales.

Por ello, la implementación de un sistema automatizado mediante técnicas de *machine learning* no solo optimizaría este proceso, sino que también proporcionaría una herramienta objetiva y eficiente para su análisis. De esta forma, se limita la capacidad de proporcionar retroalimentación oportuna y efectiva a los presentadores para que puedan mejorar sus habilidades de presentación (Quiao et al., 2017).

En este trabajo se propone un sistema de evaluación de lenguaje corporal en presentaciones orales, que utiliza algoritmos de *machine learning* para detectar y valorar la postura corporal de los presentadores (Baltrusaitis et al., 2018; Bhatt et al., 2023). El objetivo último de este sistema es habilitar una eventual retroalimentación objetiva que ayude a los presentadores a mejorar su lenguaje corporal y, en consecuencia, su comunicación oral (Hanani, et al., 2017). El objetivo de esta investigación es desarrollar un sistema que permita el reconocimiento de posturas y su evaluación en presentaciones orales mediante técnicas de aprendizaje

automático. Este sistema busca ser una herramienta que permita tanto al estudiante como al profesor tener una evaluación cuantificable de las posturas que ha realizado durante su presentación, proporcionando una herramienta precisa y eficiente para la evaluación de la comunicación no verbal (Caridakis et al., 2007; Noda, et al., 2014). La metodología para alcanzar este objetivo incluirá el diseño de algoritmos de *machine learning* que puedan procesar y analizar datos de video, identificando patrones de lenguaje corporal y correlacionándolos con criterios de evaluación establecidos (Zheng et al., 2023).

Para justificar la necesidad de este nuevo sistema, es importante destacar las deficiencias de las soluciones previas. Aunque existen herramientas que utilizan el aprendizaje automático para evaluar presentaciones orales, muchas de ellas se centran únicamente en aspectos parciales del lenguaje corporal o requieren configuraciones de hardware especializadas que limitan su accesibilidad y practicidad (Meng et al., 2020). Además, la mayoría no aborda de manera integral la interpretación del lenguaje corporal en el contexto de la comunicación humana, lo que puede resultar en evaluaciones incompletas o sesgadas (Terhürne et al., 2022).

El sistema propuesto en este proyecto pretende ser una herramienta que permita evaluar el lenguaje corporal, facilitando esta laboriosa tarea, con la finalidad de poder ser integrable con otros sistemas multimodales para la evaluación de presentaciones orales (Barua et al., 2023). Además, se prestará especial atención a las cuestiones éticas y de privacidad, implementando protocolos para proteger los datos de los usuarios y asegurar el consentimiento informado (Cummings et al., 2018).

De allí que, la investigación se enfocará en el desarrollo de un sistema integral que no solo mejore la evaluación del lenguaje corporal en presentaciones orales, sino que también respete los estándares en términos de accesibilidad y respeto por la privacidad y la ética (Mittelstadt et al., 2016). Este enfoque innovador tiene el potencial de contribuir a una mejora de las habilidades de comunicación, beneficiando tanto a los presentadores como a los evaluadores en diversos contextos educativos y profesionales (Jia, 2020).

Este trabajo de investigación se basa en la premisa de que, al mejorar la comunicación no verbal en presentaciones orales, se pueden lograr avances significativos en la efectividad de

la comunicación en una variedad de contextos (Carney et al., 2010). Desde la educación hasta la toma de decisiones empresariales y políticas, este sistema puede tener un impacto en la forma en que las personas se presentan y se comunican. Así pues, al proporcionar información a un ponente sobre el lenguaje corporal que ha utilizado en una presentación, se le puede orientar de modo que pueda mejorar sus habilidades comunicativas.

Este trabajo se estructura de la manera siguiente: en primer lugar, en este capítulo se presentan los objetivos del estudio, tanto el general como los específicos, delineando el propósito y las metas a alcanzar. A continuación, se desarrolla el marco teórico que abarca desde la teoría del lenguaje corporal hasta la ética y privacidad en la evaluación automática de posturas. Se revisan también las tecnologías y herramientas utilizadas en *machine learning*, así como su aplicación en la clasificación de posturas.

Posteriormente, se detalla la metodología empleada para llevar a cabo la investigación, seguida de la presentación de los resultados obtenidos. En el capítulo de discusión se analizan y contextualizan los hallazgos, mientras que las conclusiones consolidan los principales resultados y reflexiones del estudio. Por otro lado, se plantean las limitaciones conocidas de este trabajo, así como las líneas de trabajo propuestas para atender los retos pendientes o revelados a lo largo de este estudio. Finalmente, se incluye una lista de referencias bibliográficas que respaldan el trabajo realizado, junto con una serie de apéndices que aporten información complementaria. Hay que apuntar que tanto el código como los conjuntos de datos utilizados se pueden encontrar como material electrónico complementario[1].

La justificación para realizar un estudio sobre el reconocimiento del lenguaje corporal en presentaciones orales mediante técnicas de *machine learning* se sustenta en varias bases teóricas y científicas. Primero, el lenguaje corporal es un componente integral de la comunicación humana que complementa la comunicación verbal y transmite información significativa sobre las intenciones, emociones y reacciones de una persona (Mehrabian, 2017; Pease & Pease, 2019).). Por ello, la capacidad de analizar y comprender el lenguaje corporal

[1] https://bit.ly/3XLnmh7

puede mejorar la interacción humana y facilitar la comunicación no verbal (Gunes & Schuller, 2013).

En el contexto de las presentaciones orales, el lenguaje corporal juega un papel crucial en la efectividad del orador para transmitir su mensaje y mantener la atención de la audiencia (McNeill, 2005). Por lo tanto, un sistema que pueda reconocer y analizar el lenguaje corporal tiene cierto potencial de ofrecer retroalimentación valiosa para mejorar las habilidades de presentación (Knapp, 2013).

Además, las técnicas de *machine learning* ofrecen un enfoque robusto y escalable para el análisis de datos complejos, como los patrones de movimiento humano, lo que permite un reconocimiento preciso y en tiempo real del lenguaje corporal (Goodfellow, 2016). La investigación en este campo puede apoyarse en estudios previos que han demostrado la eficacia de las técnicas de *machine learning* en el reconocimiento de gestos y posturas corporales, con resultados que alcanzan una precisión cercana al 100% en algunos casos (Jegham et al., 2020: Pham et al., (2022)). Estos avances tecnológicos, como el uso de dispositivos de interacción natural, han abierto nuevas posibilidades para interfaces más intuitivas y expresivas entre humanos y máquinas (Zhu et al. 2013).

Además, la justificación teórica se refuerza con la necesidad de desarrollar sistemas que puedan adaptarse a diferentes contexturas físicas y destrezas de los usuarios, lo que es posible gracias a la flexibilidad de las técnicas de *machine learning* (Ojeda-Castelo, 2022). Estas técnicas pueden aprender de un conjunto etiquetado de gestos de ejemplo y luego identificar nuevos gestos como uno de los aprendidos, lo que es especialmente útil en entornos donde la variabilidad humana es significativa (Wu, 2024).

Finalmente, la justificación técnica se refuerza por la contribución potencial de este estudio al campo de la inteligencia artificial y la interacción humano-computadora, proporcionando un marco de referencia para futuras investigaciones y aplicaciones prácticas en diversos campos, como la educación, la psicología y la seguridad (Baltrusaitis et al., 2018). En resumen, la realización de este estudio está justificada por su relevancia teórica, así como por su potencial científico-técnico y su potencial para impactar positivamente en la sociedad y la tecnología (Pan & Yang, 2016).

El porqué de este estudio se justifica por su potencial para mejorar la evaluación automática y objetiva del lenguaje corporal en presentaciones orales. Este avance tiene aplicaciones en diversos campos, incluyendo la educación, la psicología, y la seguridad, al proporcionar un marco de referencia para futuras investigaciones y aplicaciones prácticas.

Específicamente en términos de seguridad, el reconocimiento del lenguaje corporal puede ser crítico para identificar comportamientos sospechosos o inusuales en entornos de alto riesgo, como aeropuertos, eventos masivos, o instalaciones sensibles. La capacidad de analizar patrones de movimiento y postura de manera precisa y en tiempo real permite la detección temprana de amenazas potenciales, contribuyendo así a la prevención de incidentes y a la mejora de las medidas de seguridad. Esta tecnología también podría ser integrada en sistemas de vigilancia para monitorear la interacción humana y responder a comportamientos que puedan indicar peligro o comportamiento no autorizado.

Además, el uso de algoritmos de *machine learning* para analizar y comprender el lenguaje corporal no solo mejora la precisión y efectividad en la evaluación, sino que también reduce la dependencia de los juicios subjetivos de los evaluadores humanos, lo cual puede ser susceptible a sesgos. Esta objetividad incrementada puede ser crucial en la toma de decisiones críticas en contextos de seguridad.

1.1. Objetivos

En esta sección se establecen las metas y propósitos fundamentales que guían el desarrollo del estudio, facilitando la consecución de los resultados deseados y el logro de conclusiones significativas.

Objetivo general

Desarrollar un sistema de reconocimiento y evaluación del lenguaje corporal en presentaciones orales utilizando técnicas de *machine learning*.

Objetivos específicos

- *Objetivo Específico 1: Identificar y recopilar datos de entrenamiento en donde se puedan registrar las* posturas *corporales de los oradores:* Al tener acceso a datos que registran las posturas corporales de los oradores durante las presentaciones, se pueden construir conjuntos de datos representativos que permitan entrenar algoritmos de *machine learning* de manera efectiva.

- *Objetivo Específico 2: Analizar los principios de visión por computadora y las técnicas para detectar y seguir los puntos de articulación del cuerpo en imágenes/videos:* El cumplimiento de este objetivo proporciona el fundamento técnico necesario para el desarrollo del sistema de reconocimiento y evaluación.

- *Objetivo Específico 3: Definir y aplicar los criterios para clasificar qué constituye una pose "positiva" o "negativa" en el contexto de una presentación oral:* Estos criterios proporcionan una guía clara para la identificación de posturas relevantes y contribuyen a la objetividad del proceso de evaluación.

- *Objetivo Específico 4: Aplicar técnicas de machine learning para entrenar modelos que puedan clasificar las* posturas *detectadas en función de los criterios establecidos, y evaluar los resultados obtenidos:* Esto permite automatizar el proceso de evaluación del lenguaje corporal en presentaciones orales, y al evaluar los resultados obtenidos, se puede verificar la efectividad y precisión del sistema desarrollado, validando así su utilidad para el propósito previsto.

1.2. Metodología

El desarrollo del sistema propuesto se basa en el uso de *MediaPipe*, que como se explicó en la introducción es, una herramienta de aprendizaje automático que permite el reconocimiento de la posición de las manos, la cara y otras partes del cuerpo humano en imágenes y videos. Así mismo, conviene explicar que se hizo uso de *datasets* que se refieren a conjuntos de datos estructurados que se utilizan para entrenar y evaluar modelos de aprendizaje automático. En

el contexto del reconocimiento del lenguaje corporal y la clasificación de poses, los *datasets* consisten en imágenes y videos etiquetados que representan diversas posturas humanas. Estos conjuntos de datos son fundamentales para enseñar a los algoritmos a identificar y clasificar correctamente los gestos y posturas como "positivas" o "negativas", basándose en criterios predefinidos (Kotsiantis et al., 2007).

El objetivo del proyecto es analizar el lenguaje corporal de los sujetos que realizan presentaciones orales y clasificar sus posturas como "positivas" o "negativas" según unos criterios preestablecidos, de acuerdo con una serie de pasos que se plasman en la memoria de la siguiente forma:

1. **Estado del arte:** En este capítulo, se incluye un párrafo que menciona que existen varios sistemas que emplean aprendizaje automático para la evaluación de competencias orales. Se plantea el objetivo general del proyecto, que es analizar el lenguaje corporal en presentaciones orales y clasificar las posturas como "positivas" o "negativas".

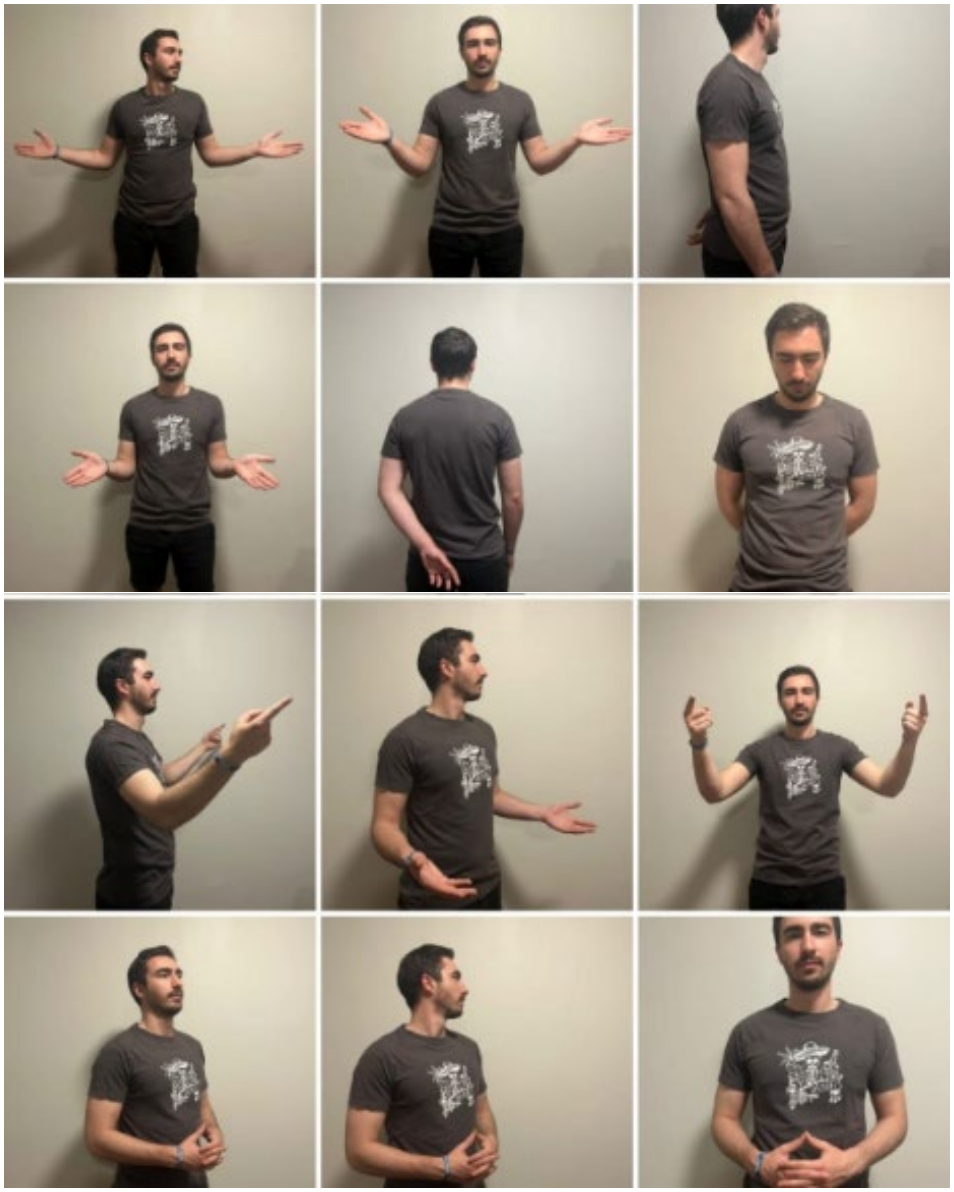
2. **Metodología detallada:** En este capítulo del proyecto, se presenta una metodología más detallada, que incluye discusiones sobre el método experimental adoptado, la muestra de conveniencia utilizada y su impacto en la generalización, los datos de la muestra, los pasos experimentales y los resultados obtenidos. Este capítulo proporciona información más específica sobre la implementación del proyecto y los detalles del proceso. Es importante seguir estos pasos para llevar a cabo el proyecto de manera efectiva y obtener resultados confiables en el análisis del lenguaje corporal en presentaciones orales.

3. **Definición de posturas:** Se han definido una serie de posturas que se consideran indicativas de un buen o mal desempeño en una presentación oral. Por ejemplo, una pose positiva podría ser tener las manos abiertas y los brazos extendidos, mientras que una pose negativa podría ser cruzar los brazos o encogerse de hombros. Cada imagen se etiqueta con la categoría "positiva" o "negativa" según la pose realizada. Es importante tener en cuenta el contexto cultural de la persona que realiza el gesto,

ya que un mismo gesto puede interpretarse de diversas formas según la cultura. Además, se registran los ángulos y condiciones de captura para futuras referencias. Unos ejemplos de posturas empleadas en el estudio se pueden ver en la ***Figura 1***:

Figura 1

Ejemplos de posturas para el dataset de entrenamiento



Fuente: Elaboración propia.

4. **Captura de imágenes:** Se dispone de un grupo de 26 voluntarios, compuesto por 12 hombres y 11 mujeres, con edades entre 22 y 35 años, estudiantes y profesionales universitarios. A cada participante se le indica que tome una postura específica, y se le capten cuatro fotografías por postura: de frente, de espaldas y de perfil derecho. Se realizan tareas de preprocesamiento en las imágenes, como ajuste de tamaño, normalización de colores y recorte, para garantizar la uniformidad y calidad de la base de datos. La ***Tabla 1*** muestra una esquematización del proceso de captura de imágenes:

Tabla 1

Posturas para el dataset de entrenamiento del modelo

Etiqueta	Posición del cuerpo respecto a la cámara	Descripción	Clasificación
Brazos abajo o caídos	Frente Lado izquierdo Lado derecho De espaldas	Expresa poca emocionalidad, poca energía	Negativa
Brazos a 90°	Frente Lado izquierdo Lado derecho De espaldas	Es una posición expansiva del cuerpo, que denota seguridad, confianza y momento de explicación del tema	Positiva
Manos en los bolsillos	Frente Lado izquierdo Lado derecho De espaldas	En esta pose se busca ocultar las manos o "protegerlas" de la vista de la audiencia	Negativa
Manos detrás del cuerpo	Frente Lado izquierdo Lado derecho De espaldas	En esta pose se busca ocultar las manos o "protegerlas" de la vista de la audiencia	Negativa
Brazos cruzados	Frente Lado izquierdo Lado derecho De espaldas	En esta pose se busca ocultar las manos o "protegerlas" de la vista de la audiencia	Negativa
Brazo derecho agarrando el brazo izquierdo por el frente del cuerpo	Frente Lado izquierdo Lado derecho De espaldas	En esta pose se busca ocultar las manos o "protegerlas" de la vista de la audiencia	Negativa
Brazo izquierdo agarrando el brazo derecho por el frente del cuerpo	Frente Lado izquierdo Lado derecho De espaldas	En esta pose se busca ocultar las manos o "protegerlas" de la vista de la audiencia	Negativa

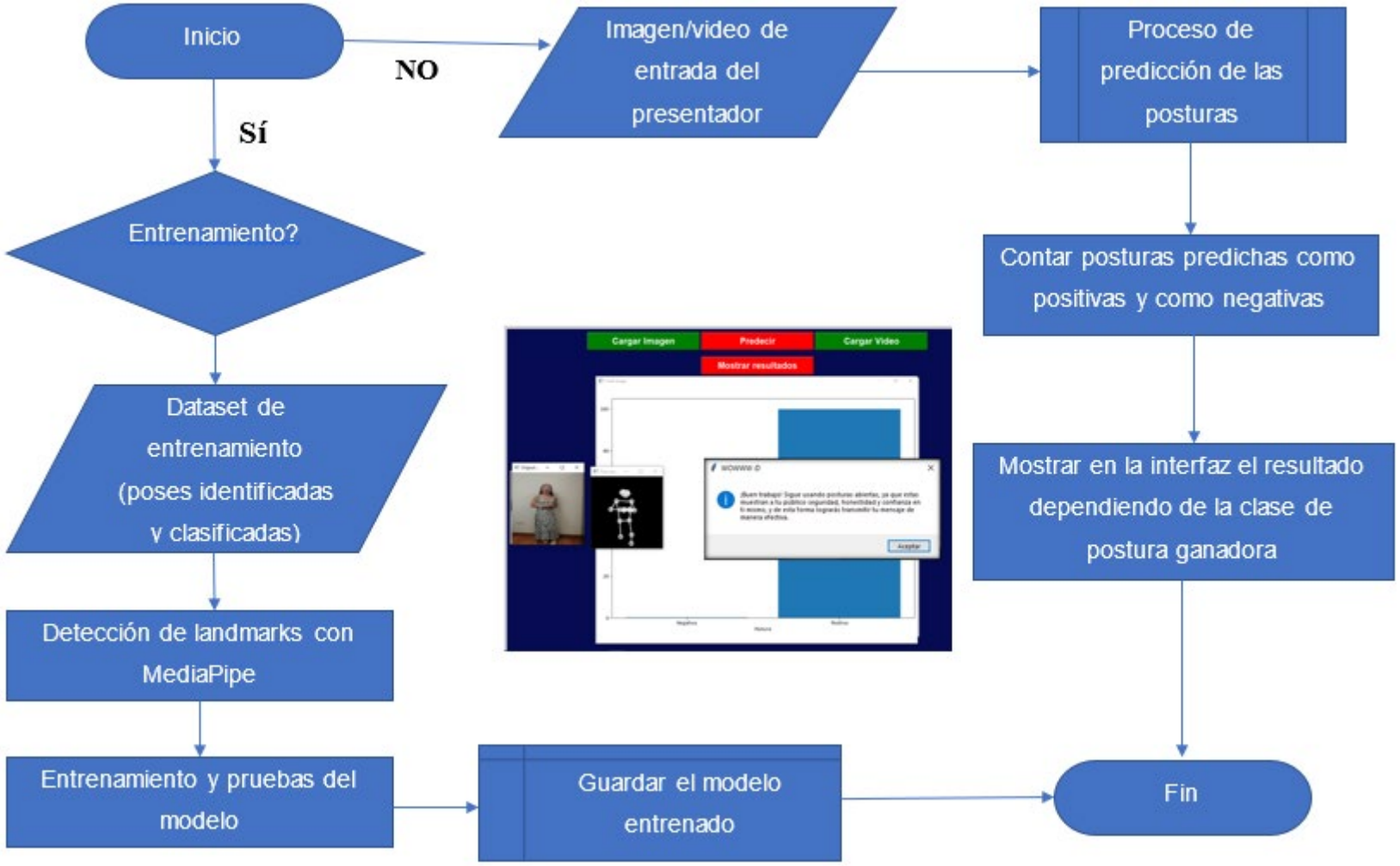
Manos con las palmas abiertas y de frente	Frente Lado izquierdo Lado derecho De espaldas	Se busca mostrar las partes blandas del cuerpo o ventralidad con el fin de expresar cercanía con el público, confianza	Positiva
Manos mostrando el dorso hacia el frente	Frente Lado izquierdo Lado derecho De espaldas	Se busca mostrar las partes blandas del cuerpo o ventralidad con el fin de expresar cercanía con el público, confianza	Positiva
Manos entrelazadas	Frente Lado izquierdo Lado derecho De espaldas	Es un gesto que denota seguridad	Positiva

Fuente: Elaboración propia a partir de Castro (2023) y Lubienetzki & Schüler-Lubienetzki (2022).

5. **Procesamiento con *MediaPipe*:** Las imágenes se procesan con *MediaPipe* para obtener las coordenadas de los puntos clave del cuerpo humano (*landmarks*) y se almacenan en una base de datos junto con la etiqueta de la pose correspondiente. En esta base de datos se disponen conjuntos de entrenamiento y prueba para evaluar la precisión del modelo durante el proceso de entrenamiento. En la ***Figura 2*** se ilustra visualmente este proceso y los pasos descritos anteriormente.

Figura 2

Diagrama de bloques del proceso



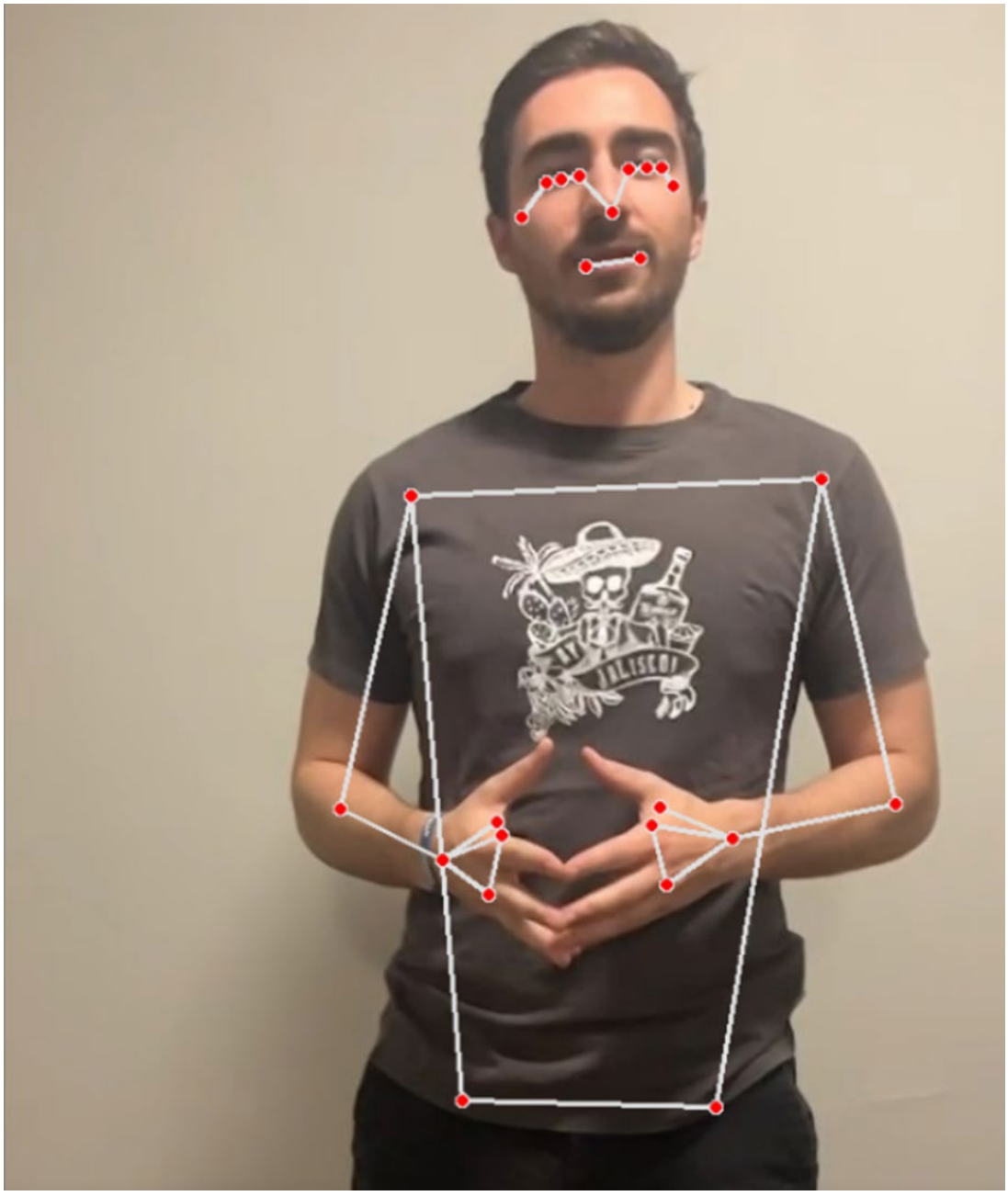
Fuente: Elaboración propia

En esta fase del proceso, se emplea *MediaPipe* para identificar puntos clave de articulación en el cuerpo, como las manos, el rostro y otras áreas significativas. Mediante el análisis de video con *MediaPipe*, se extraen las coordenadas de los puntos de referencia del cuerpo humano en momentos específicos, por ejemplo, cada segundo. Estas coordenadas se cotejan con una base de datos para asignar la postura más similar en cada intervalo de tiempo. Posteriormente, el sistema analiza la postura de los participantes en imágenes capturadas y evalúa la cantidad de posturas positivas vs la cantidad de posturas negativas, arrojando como resultado un mensaje de recomendación si la presentación tenía más posturas negativas o un mensaje de felicitación si la presentación contenía más posturas positivas que negativas.

6. **Detección de *landmarks:*** En la cuarta fase, se mide la calidad de la presentación oral de los participantes. Se busca establecer una correlación entre las posturas positivas y la calidad de la presentación, así como entre las posturas negativas y una menor calidad de la misma. Se anticipa que este análisis revele una conexión directa entre la expresión corporal y la efectividad de la comunicación oral. Cabe mencionar que los *landmarks* son puntos clave específicos en el cuerpo humano que se utilizan para identificar y seguir partes importantes del cuerpo, como las articulaciones, en imágenes y videos. Estos puntos clave son detectados y ubicados mediante algoritmos de visión por computadora y aprendizaje automático, permitiendo así el análisis detallado de las poses y movimientos corporales. En el contexto del reconocimiento del lenguaje corporal, los *landmarks* son esenciales para mapear y clasificar las posturas como "positivas" o "negativas", proporcionando una representación estructurada de la postura del cuerpo. En el análisis del lenguaje corporal, estos puntos clave son detectados automáticamente utilizando herramientas como *MediaPipe*. Posteriormente, se analizan para clasificar las posturas de los individuos en categorías como "positivas" o "negativas". Este proceso permite obtener una evaluación detallada y objetiva del lenguaje corporal en presentaciones orales y otras aplicaciones. (Cao et al., 2021). La ***Figura 3*** muestra ejemplos de detección de *landmarks* con *MediaPipe*:

Figura 3

Ejemplos de detección de landmarks con MediaPipe



Fuente: Elaboración propia.

2. Estado del arte

El estudio actual se encuentra en la intersección de la comunicación no verbal y la inteligencia artificial. Por ello el estudio se enfoca en el desarrollo de algoritmos capaces de interpretar y analizar el lenguaje corporal humano, lo que constituye un área de gran interés debido a su aplicabilidad en diversos contextos, desde la educación hasta la negociación empresarial (Gunes & Schuller, 2013: Pantic et al., 2005; Vinciarelli et al., 2009). La utilización de técnicas de *machine learning* permite que el sistema aprenda de manera autónoma a reconocer patrones y gestos, mejorando su precisión con el tiempo a partir del eventual incremento de los datos de entrenamiento (Jegham et al., 2020).

Este enfoque innovador promete transformar la manera en que entendemos y evaluamos las presentaciones orales, proporcionando herramientas para mejorar la comunicación y la expresión personal (Baltrusaitis et al., 2018). La investigación actual se basa en un corpus de estudios previos que han explorado desde los fundamentos teóricos del lenguaje corporal hasta aplicaciones prácticas de la inteligencia artificial, estableciendo un marco sólido para futuras investigaciones y desarrollos tecnológicos en este ámbito (Caridakis et al., 2007; Pantic et al., 2005). Un somero análisis de dichos estudios previos se presenta en las siguientes secciones de este capítulo.

2.1. Teoría del lenguaje corporal

La teoría del lenguaje corporal se basa en el principio de que la comunicación humana no se limita a las palabras habladas o escritas, sino que involucra una serie de señales no verbales, como gestos, posturas, expresiones faciales y movimientos, que transmiten información adicional. Estas señales no verbales pueden ser tan importantes, o incluso más, que las palabras mismas a la hora de comunicar emociones, actitudes y mensajes en general (Mehrabian, 1981). Esta teoría proporciona un marco fundamental para comprender la relevancia del análisis del lenguaje corporal en la evaluación de presentaciones orales.

Mehrabian (1981) es considerado un autor clave en el campo de la comunicación no verbal, debido al efecto de su investigación en otros trabajos de este ámbito (Petty & Cacioppo, 1986; Holland et al., 2017; Kaur et al., 2020; Lubienetzki & Schüler-Lubienetzki, 2022). Su

investigación sugiere que, en situaciones donde el contenido de un mensaje es ambiguo o contradictorio con la comunicación no verbal, la gente tiende a confiar en las señales no verbales para interpretar la comunicación. De acuerdo con sus estudios, el lenguaje corporal y las expresiones faciales representan aproximadamente el 55% de la comunicación, mientras que el tono de voz contribuye con el 38%, y las palabras en sí mismas solo con el 7%. Esto subraya la importancia del lenguaje corporal en la interpretación de las intenciones y emociones de un orador durante una presentación oral.

Al trasladar la teoría del lenguaje corporal al contexto de la evaluación de posturas en presentaciones orales, se puede extender la idea de que analizar gestos y posturas puede proporcionar información valiosa sobre el desempeño del presentador y su efectividad en la comunicación con la audiencia. Un sistema de detección mediante visión por computador puede capturar información con un nivel de detalle casi imposible para un ser humano (Gulati, 2022). Considerando la importancia de los gestos, un ejemplo podría ser un presentador quien, con una pose abierta y gestos positivos puede transmitir confianza y entusiasmo, mientras que un presentador con posturas cerradas y gestos negativos puede transmitir falta de confianza o nerviosismo.

Además, la teoría del lenguaje corporal también destaca que las señales no verbales pueden variar en su interpretación según el contexto cultural y situacional. Por lo tanto, es esencial tener en cuenta la diversidad cultural al desarrollar algoritmos de evaluación de posturas para que sean aplicables en diferentes contextos y audiencias.

Por otra parte, Carney et al. (2010), en su investigación, exploraron el concepto de "*power posing*", que se refiere a adoptar posturas abiertas y expansivas que reflejan confianza y dominio. Los resultados del estudio mostraron que las personas que adoptaron estas posturas durante tan solo dos minutos experimentaron cambios en los niveles de hormonas, específicamente un aumento en los niveles de testosterona y una disminución en los niveles de cortisol, hormonas asociadas con la confianza y el estrés, respectivamente.

La relevancia del estudio de (Carney et al., 2010), para la teoría del lenguaje corporal y la evaluación de posturas o posturas en presentaciones orales, radica en que sugiere que las posturas o posturas corporales no solo son indicadores visuales de la comunicación no verbal,

sino que también pueden tener un impacto biológico en el estado emocional y la confianza de un individuo. Por lo tanto, si un presentador adopta posturas positivas y de poder durante una presentación, no solo puede transmitir confianza a la audiencia, sino que también podría experimentar un aumento en su propia confianza y reducción del estrés, lo que puede mejorar su desempeño.

Así pues, el análisis de la teoría del lenguaje corporal subraya la importancia de las posturas corporales en la comunicación no verbal. Este fundamento teórico destaca que las posturas no solo comunican emociones y actitudes, sino que también pueden influir en el estado emocional y la confianza del presentador. El análisis de la teoría del lenguaje corporal enfatiza la trascendencia de las posturas corporales en la comunicación no verbal. La relevancia de desarrollar algoritmos para la evaluación automática de posturas en presentaciones orales se sustenta en la necesidad de complementar los métodos de evaluación tradicionales, como las escalas de puntaje o las rúbricas (Rayón, Guenaga & Núñez, 2014). Mediante el empleo de técnicas de detección de posturas, se pueden construir sistemas que (i) proporcionen información de aquellos que han sido usados, con mayor frecuencia, por un ponente, durante un discurso o presentación, y que permitan que (ii) esta información se transformase en retroalimentación que, a su vez, pueda contribuir a la mejora de las habilidades de presentación y la eficacia de la comunicación.

2.2. Psicología de la comunicación

La comprensión de la teoría del lenguaje corporal proporciona la base para adentrarse en el estudio de la psicología de la comunicación. La teoría del lenguaje corporal examina cómo los gestos, expresiones faciales y posturas corporales influyen en la percepción y comprensión durante las interacciones comunicativas. Este conocimiento es fundamental para comprender cómo el lenguaje corporal contribuye a la efectividad de la comunicación. Al explorar la psicología de la comunicación, se profundiza en los procesos mentales y emocionales que subyacen a la interacción humana, incluyendo la forma en que se procesa la información, se construyen las percepciones y se establecen relaciones sociales.

La psicología de la comunicación es un campo interdisciplinario que investiga cómo las personas procesan, interpretan y responden a los mensajes en diversos contextos. Esta

disciplina es especialmente relevante en la evaluación de la comunicación no verbal en presentaciones orales, ya que proporciona un marco teórico para comprender cómo el lenguaje corporal puede influir en la percepción del orador y la efectividad de su mensaje. Los siguientes cuatro elementos clave ayudan a explicar la relación entre el procesamiento de la información, la persuasión, la percepción y la influencia de la comunicación no verbal en este contexto (Shu et al., 2020).

Procesamiento de la información: La teoría del procesamiento de la información se centra en cómo las personas perciben y retienen la información. En presentaciones orales, esto implica cómo los espectadores asimilan las señales no verbales, como gestos y posturas. Un lenguaje corporal coherente y eficaz facilita la comprensión y la memoria de la audiencia (Jaramillo et al., 2013).

Teoría de la persuasión: La teoría de la persuasión se enfoca en cómo influir en la audiencia. En presentaciones orales, el lenguaje corporal desempeña un papel crucial en la persuasión, influyendo en la aceptación del mensaje por parte de la audiencia. Modelos como el de probabilidad de elaboración explican cómo gestos y posturas pueden impactar en la persuasión (Holland et al., 2017; Petty y& Cacioppo, 1986).

Percepción y empatía: La percepción del orador por parte de la audiencia es esencial. Los gestos y posturas influyen significativamente en cómo se percibe al orador, y un lenguaje corporal empático puede fortalecer la conexión con la audiencia, facilitando la empatía y la receptividad del mensaje (Cialdini, 2021).

Influencia de la comunicación no verbal: Investigaciones han demostrado que la comunicación no verbal puede tener mayor influencia que las palabras habladas. Gestos, posturas y expresiones faciales juegan un papel clave en cómo se recibe e interpreta el mensaje, y su eficacia puede variar según el contexto cultural, lo que exige un enfoque sensible y adaptativo en la evaluación del lenguaje corporal (Carney, 2010; Cialdini, 2021; Lubienetzki & Schüler-Lubienetzki, 2022).

En resumen, la psicología de la comunicación ofrece un marco integral para entender el impacto del lenguaje corporal en la efectividad de las presentaciones orales. El procesamiento

de la información, la persuasión, la percepción y la comunicación no verbal son elementos clave que interactúan para moldear cómo los mensajes son recibidos y comprendidos por la audiencia. Este conocimiento es esencial para diseñar estrategias de comunicación más efectivas y conscientes de la diversidad cultural.

2.3. Teoría de la persuasión y la efectividad de la presentación

Como ya se ha dicho, la teoría de la persuasión desempeña un papel crítico en la evaluación de posturas en presentaciones orales, ya que se enfoca en cómo los oradores pueden influir en la audiencia y lograr sus objetivos de comunicación (Cialdini, 2021). A continuación, se expone el modo en que esta teoría se relaciona con la efectividad de la presentación y la evaluación automática de posturas:

1. *Centralidad de la Ruta vs. Periférica:* La teoría de la persuasión distingue entre dos rutas para la persuasión: la central y la periférica. La ruta central implica argumentos sólidos y evidencia convincente, mientras que la ruta periférica se basa en factores periféricos, como la apariencia y el lenguaje corporal del orador (Petty & Cacioppo, 1986; Cialdini, 2021). El lenguaje corporal efectivo puede influir en la persuasión periférica (Lubienetzki & Schüler-Lubienetzki, 2022), lo que resalta la importancia de evaluar posturas en presentaciones.

2. *Atractividad y Credibilidad del Orador:* La teoría de la persuasión enfatiza la importancia de la credibilidad y la atractividad del orador. Un presentador con posturas y gestos confiables y atractivos es más persuasivo. Evaluar la postura y el lenguaje corporal puede ayudar a determinar la credibilidad y la atractividad del orador (Holland et al., 2017).

3. *Efecto Halo:* El efecto halo sugiere que las personas generalizan sus impresiones de una característica positiva de un orador a otras características. Por lo tanto, si un presentador proyecta una postura positiva y segura, es más probable que se le atribuyan otras cualidades positivas (Verhulst et al., 2010). La evaluación de posturas puede identificar el efecto halo en las presentaciones.

4. *Empatía y Relación con la Audiencia:* La teoría de la persuasión resalta la importancia de establecer una relación empática con la audiencia. Los gestos y posturas empáticas pueden influir en la percepción de la audiencia y su disposición para ser persuadida (Holland et al., 2017).

5. *Mensaje y Audiencia Dirigida:* La persuasión eficaz implica adaptar el mensaje y la presentación a la audiencia (Cialdini, 2021). La evaluación automática de posturas puede ayudar a identificar cómo un presentador se adapta a su audiencia mediante el análisis de su lenguaje corporal.

6. *Teoría de la Disonancia Cognitiva:* La teoría de la disonancia cognitiva sostiene que las personas buscan alinear sus creencias y actitudes. Las posturas contradictorias o incongruentes con el mensaje verbal pueden generar disonancia cognitiva (Chen & Risen, 2010). Evaluar las posturas puede identificar tales incongruencias en las presentaciones.

7. *Motivación del Cambio:* La teoría de la persuasión también se relaciona con la motivación para el cambio (Cialdini, 2021). La postura y el lenguaje corporal pueden influir en la motivación de la audiencia para adoptar nuevas actitudes o comportamientos.

2.4. Trabajos previos relacionados

La revisión de trabajos previos relacionados es esencial en cualquier proyecto de investigación, ya que permite comprender el estado actual del campo y establecer una base sólida para el nuevo enfoque propuesto. En el caso de la evaluación automática de posturas en presentaciones orales, existen investigaciones previas que han abordado temas relacionados, a continuación, se presentan algunas de ellas:

1. *Behoora & Tucker (2015):* El estudio se enfocó en la detección de estados emocionales en equipos de diseño, utilizando el lenguaje corporal como indicador. Entre sus hallazgos se encuentra que el uso de *machine learning* demuestra la capacidad de las técnicas de aprendizaje automático para analizar datos de lenguaje

corporal y detectar patrones que pueden ser difíciles de identificar manualmente. Además, el comprender cómo el lenguaje corporal influye en la percepción de las emociones puede ayudar a los presentadores a mejorar sus habilidades, ya que pueden utilizar esta información para ajustar su lenguaje corporal y transmitir sus emociones de manera más efectiva, lo que puede resultar en presentaciones más impactantes.

2. *Chen et al. (2014):* El trabajo se centró en la evaluación automatizada de habilidades de hablar en público utilizando señales multimodales, que incluyen lenguaje corporal. El estudio emplea técnicas de aprendizaje automático para clasificar las posturas observadas y evaluar las habilidades de presentación, aplicables al desarrollo de un sistema de evaluación de posturas, ya que las redes neuronales de convolución y otros algoritmos de aprendizaje automático pueden utilizarse para analizar el lenguaje corporal en el momento en que se realiza la presentación oral. También los autores sugieren que esta evaluación automatizada pudiese tener aplicaciones en el ámbito educativo, donde la retroalimentación automatizada puede ayudar a los estudiantes a mejorar sus habilidades de presentación.

3. *Wörtwein et al. (2015):* En su investigación exploraron el uso de señales multimodales para evaluar el desempeño en presentaciones orales. El trabajo se centra en el desarrollo de un sistema de evaluación automatizado de presentaciones, lo que proporciona una metodología y un marco de referencia para el proyecto en cuestión. Por otra parte, el estudio describe la extracción de características y el uso de técnicas de aprendizaje automático para evaluar las presentaciones, en conjunto con la construcción de una evaluación objetiva y consistente de las habilidades de presentación.

4. *Schneider et al. (2015):* Los autores investigaron la creación de un "entrenador de presentaciones" que utiliza señales multimodales para proporcionar retroalimentación en tiempo real. La importancia de este trabajo radica en su contribución al desarrollo de sistemas de reconocimiento del lenguaje corporal en presentaciones orales utilizando técnicas de *machine learning*. Al proporcionar retroalimentación

inmediata sobre la postura corporal, los movimientos y la voz del usuario, el "entrenador de presentaciones" puede ayudar a los usuarios a mejorar sus habilidades de presentación y oratoria.

5. *Holland et al. (2017):* En el estudio los investigadores exploraron cómo las posturas corporales influyen en la atención visual. Este trabajo destaca la importancia de las posturas en la percepción de la audiencia, demostrando que las personas tienden a desviar la mirada de las posturas que transmiten dominancia no verbal. Este fenómeno sugiere que las posturas de poder pueden provocar respuestas automáticas de evitación en el espectador, lo que resalta la importancia del lenguaje corporal en la percepción social. Los hallazgos del estudio tienen implicaciones importantes para la comprensión de cómo el lenguaje corporal puede afectar la interacción social, especialmente en contextos donde la percepción de autoridad y poder es relevante, como en entornos laborales y de liderazgo. En este sentido, la investigación contribuye a la literatura sobre comunicación no verbal al mostrar cómo las señales de dominancia pueden impactar la atención y el comportamiento de los demás, subrayando la necesidad de ser conscientes de las posturas adoptadas en situaciones de interacción social para influir en la dinámica de la comunicación y la percepción de la audiencia.

6. *Ochoa (2022):* En este capítulo de libro los investigadores sugieren la importancia de desarrollar habilidades de comunicación oral en los estudiantes, debido a que en la sociedad moderna son esenciales para el éxito empresarial, académico etc. Sin embargo, proporcionar retroalimentación constante y efectiva en las presentaciones orales es un desafío para los profesores dada la carga de trabajo que esto conlleva. Los investigadores realizan una comparativa con varios sistemas *OPAF (Sistemas de Retroalimentación Automatizada de Presentaciones Orales)* desarrollados para mejorar las habilidades de presentación oral mediante retroalimentación automatizada. De acuerdo a sus puntos clave del análisis comparativo de estos sistemas, se puede concluir lo siguiente: aunque hay una variedad de sistemas *OPAF*, la mayoría utiliza técnicas relativamente básicas para analizar y retroalimentar las

> habilidades de presentación oral, con pocos avances significativos en métodos más sofisticados como el aprendizaje automático. Este capítulo invita a los investigadores de sistemas *OPAF* a desarrollar sistemas que sean pedagógicamente útiles, que se enfoquen en habilidades específicas, en lugar de generales, utilizando diferentes tipos de retroalimentación.

Entonces, el estudio realizado por Ochoa (2022), se enfoca en la evaluación y comparación de varios sistemas de retroalimentación automatizada para presentaciones orales (OPAF). En el análisis, se consideran cinco sistemas existentes que utilizan diferentes enfoques para proporcionar retroalimentación a los presentadores. A continuación, se presenta la ***Tabla 2*** generada por Ochoa en su estudio, que sintetiza las características clave de cada sistema analizado:

Tabla 2

Características de cada sistema analizado

Nombre del Sistema	Año	Referencia	Aportes Clave	Enfoque Tecnológico	Modalidades Extraídas	Intrusividad	Tipo de retro-alimentación	Desventajas
Presentation Sensei	2007	Kurihara et al. (2007)	Pionero en el uso de visión por ordenador y procesamiento del habla para la formación en presentaciones orales.	Procesamiento de imagen y voz	Lenguaje corporal, Volumen, Tono	Media	En tiempo real	Intrusivo debido a los sensores utilizados.
Cicero	2013	Batrinca et al. (2013)	Uso de audiencia virtual interactiva para simular una retroalimentación realista.	Audiencia virtual interactiva	Postura, Gestos, Volumen, Tono	Media	Evaluación visual y auditiva	Sistema más complejo, alta necesidad de procesamiento computacional.
Logue	2015	Damian et al. (2015)	Utiliza Google Glass para obtener vistas egocéntricas y mejorar la calidad de las presentaciones orales.	Google Glass y sensores ópticos	Vistas egocéntricas, Contacto visual	Alta	En tiempo real	Requiere equipo especializado, alta intrusividad.
[Dermondy]	2015	Dermondy y Sutherland (2015)	Proporciona retroalimentación en tiempo real para mejorar el rendimiento del presentador.	Sensores multimodales y feedback real-time	Retroalimentación multimodal	Alta	En tiempo real	Uso limitado fuera de entornos controlados.
NUSMSP	2015	Gan et al. (2015)	Usa Google Glass para capturar movimientos de la cabeza y orientación del presentador.	Google Glass y sensores de movimiento	Movimientos de cabeza, Orientación	Alta	Evaluación general	Requiere equipo especializado (Google Glass), alta intrusividad.
PresentMate	2015	Lui et al. (2015)	Utiliza sensores de teléfonos móviles para capturar movimientos y voz del presentador.	Sensores móviles (acelerómetro, micrófono)	Movimientos corporales, Voz	Baja	Después de la presentación	Uso limitado en contextos que no permiten la manipulación del teléfono.
[Nguyen]	2015	Nguyen et al. (2015)	Enfocado en la evaluación automática de presentaciones.	Sensores varios y sistemas de evaluación automática	Evaluación automática	Alta	Evaluación general	Uso limitado de modalidades para capturar la presentación.
Presentation Trainer	2015	Schneider et al. (2015)	Proporciona retroalimentación sobre el desempeño del presentador utilizando Microsoft Kinect.	Microsoft Kinect	Movimiento corporal, Voz	Alta	En tiempo real	Alta intrusividad y necesidad de espacio.
Rhema	2015	Tanveer et al. (2015)	Proporciona retroalimentación en tiempo real a través de Google Glass sobre el ritmo y volumen de la presentación.	Google Glass y análisis de voz	Volumen, Ritmo	Alta	En tiempo real	Alta intrusividad debido a la necesidad de usar Google Glass.
RoboCOP	2017	Trinh et al. (2017)	Usa un robot humanoide para proporcionar retroalimentación verbal y no verbal al presentador.	Robótica interactiva y retroalimentación verbal	Postura, Gestos, Voz	Media	Verbal y no verbal	Complejidad técnica alta, difícil de implementar fuera del laboratorio.
RAP	2018	Ochoa et al. (2018)	Bajo coste e intrusividad, utiliza una cámara web y algoritmos de visión por computador para proporcionar retroalimentación sobre presentaciones orales.	Visión por computadora y análisis de voz	Postura, Contacto visual, Diapositivas	Baja	Post-presentación	Proporciona menos detalles en tiempo real, retroalimentación posterior a la presentación.
Presentation Trainer VR	2019	Schneider et al. (2019)	Utiliza realidad virtual para entrenar a los presentadores mediante retroalimentación visual y auditiva.	Realidad virtual (Hololens)	Postura, Tono, Pausas	Alta	En tiempo real y post-presentación	Alta intrusividad, requerimiento de equipo especializado como Hololens.

Fuente: Elaboración propia

La inclusión de esta tabla ofrece una visión clara y comparativa de los sistemas evaluados en el estudio de Ochoa, permitiendo destacar no solo las diferencias tecnológicas, sino también la efectividad de cada sistema en proporcionar retroalimentación. Al comparar estos sistemas, Ochoa concluye que, aunque todos ofrecen algún nivel de retroalimentación, los más avanzados combinan múltiples modos de análisis para ofrecer una retroalimentación más rica y detallada, destacando la importancia de un enfoque integral en la mejora de las habilidades de presentación.

La revisión de estos trabajos previos proporciona un contexto importante para el proyecto de evaluación automática de posturas en presentaciones orales. Las investigaciones que se han presentado abordan temas relacionados con el lenguaje corporal, la evaluación de presentaciones y la detección de emociones, lo que puede servir como punto de partida y fuente de inspiración para el desarrollo de algoritmos y sistemas efectivos de evaluación y retroalimentación.

Por otro lado, se identifican aspectos que necesitan ser abordados. Aunque se utiliza aprendizaje automático, se sugiere que la precisión de la detección de características del lenguaje corporal aún puede mejorarse, como señalan los estudios de Chen et al. (2014) y Schneider et al. (2015). Asimismo, se plantea la necesidad de clarificar si los sistemas proporcionan retroalimentación clara sobre el lenguaje corporal detectado, como se menciona en (Schneider et al., 2015), donde se destaca la falta de interpretabilidad como una limitación. Por otro lado, se cuestiona la capacidad de generalización de los sistemas, puesto que, aunque Wörtwein et al. (2015) apunta a la evaluación multimodal del desempeño en presentaciones públicas, no se profundiza si el sistema se adapta a diferentes tipos de presentaciones y oradores. Además, se destaca la ausencia de un abordaje explícito sobre la influencia del contexto en la interpretación del lenguaje corporal. Por ejemplo, una sonrisa puede tener significados distintos en una charla de *Technology, Entertainment, and Design* (TED) o una entrevista de trabajo, aspecto que no es tratado específicamente en los trabajos revisados.

El análisis de los sistemas previos de retroalimentación automatizada para presentaciones orales revela varias limitaciones importantes que no han sido completamente abordadas en

investigaciones anteriores. Estas limitaciones incluyen la falta de integración de múltiples modalidades de retroalimentación, la limitada capacidad para ofrecer *feedback* en tiempo real, y la dependencia de hardware especializado que reduce la accesibilidad y escalabilidad de los sistemas.

1. **Falta de integración multimodal**: La mayoría de los sistemas analizados, como se muestra en la tabla de (Ochoa, 2022), tienden a enfocarse en un solo tipo de *feedback* (como el análisis de voz o gestos), dejando de lado un enfoque más holístico que considere múltiples modalidades de retroalimentación de manera simultánea. Esto reduce la efectividad del *feedback* en mejorar la competencia comunicativa global del presentador.

2. ***Feedback* en tiempo real**: Algunos sistemas ofrecen retroalimentación post-evento, lo que limita la capacidad del presentador para ajustar su desempeño en el momento. La retroalimentación en tiempo real es crucial para la corrección inmediata y para la mejora continua durante la presentación misma.

3. **Dependencia de hardware especializado**: Varios sistemas requieren dispositivos específicos o sensores avanzados que pueden no estar disponibles en todos los entornos educativos o profesionales, limitando así la adopción generalizada de estas tecnologías.

Aporte del trabajo actual: Está investigación hace un análisis de los diferentes sistemas que se han creado para la evaluación de presentaciones orales, mostrando sus limitaciones y la importancia que tiene seguir desarrollando trabajos que impliquen las mejoras de las habilidades comunicativas de los estudiantes. Este trabajo pretende crear una herramienta que sea de fácil utilización, y entendimiento para los usuarios, en principio podría ser usado por estudiantes y profesores, que deseen evaluar sus posturas en su presentación. Por otra parte, también puede servir como base para las investigaciones que pretenden mejorar los sistemas de reconocimiento y seguimiento de posturas en presentaciones orales.

Este proyecto, ha sido desarrollado con la finalidad de realizar una investigación que contemple todos los pasos para la creación de un modelo de *machine learning* que pueda ser

usado para futuras investigaciones en el campo de la evaluación automática de presentaciones orales y aportar una pequeña parte en esta área de investigación.

Para ello se han utilizado tecnologías accesibles y de bajo costo, como las bibliotecas de *MediaPipe* y *OpenCV*. El sistema es capaz de proporcionar retroalimentación al usuario, y por usuario se hace referencia al estudiante, profesor o investigador que desee evaluar su postura a lo largo de su presentación, lo que le permite al presentador ajustar su desempeño de manera congruente con sus gestos, es decir que sepa reconocer las posturas positivas y negativas que ha realizado a lo largo de su presentación. Además, la utilización de tecnologías de código abierto asegura que el sistema sea escalable y accesible para una amplia gama de usuarios, desde estudiantes hasta profesionales, ya que este proyecto no hace uso de hardware especializado ni de alto costo, y ofrece a los investigadores una herramienta que puede ser útil en cualquier etapa de investigación.

Comparación y justificación del enfoque: Las investigaciones previas se han centrado en (i) tratar de abordar todas las modalidades que se pueden encontrar en una evaluación de presentaciones orales, pero (ii) sin hacer una investigación profunda que le muestre al estudiante o profesor la importancia de las posturas en una presentación oral. Sin embargo, este trabajo pretende centrarse en la investigación de una sola modalidad, es decir la evaluación de las posturas, que constituye como se ha mencionado en los capítulos anteriores, una de las partes más relevantes para mejorar significativamente la presentación del estudiante, una buena postura influye positivamente en las demás modalidades de la presentación y se ha demostrado que psicológicamente ayuda a que el estudiante tenga mayor confianza en sí mismo lo que puede impactar positivamente en la atención y el comportamiento de la audiencia.

2.5. Ética y privacidad en la evaluación automática de posturas en presentaciones orales

La evaluación automática de posturas en presentaciones orales, a través de algoritmos de procesamiento de imágenes y el uso de inteligencia artificial, plantea cuestiones éticas y de privacidad significativas: en primer lugar, la recopilación de datos visuales de individuos durante sus presentaciones puede plantear preocupaciones éticas relacionadas con el

consentimiento informado. Es esencial garantizar que los participantes estén conscientes de que sus acciones y movimientos serán registrados y utilizados con fines de evaluación (Sánchez-Mena & Martí-Parreño, 2017).

En este sentido, la primera consideración que debe tomarse en cuenta es que, antes de recopilar datos de lenguaje corporal de los presentadores, es esencial obtener su consentimiento informado (véase ***anexo A***). Los participantes deben comprender cómo se utilizarán sus datos, quién tendrá acceso a ellos y las medidas de seguridad para proteger su privacidad (Behoora & Tucker, 2015). Por otra parte, dado que se trata de un proyecto de investigación, se abre la posibilidad de que otros trabajos se basen en todo o parte de la investigación previa, inclusive los datos recopilados, por lo que estos deben ser anonimizados para evitar la identificación de los presentadores (Chen et al., 2014).

Además, de acuerdo con Wörtwein et al. (2015) las personas deben ser informadas de manera transparente sobre cómo se utilizarán sus datos de lenguaje corporal y cómo se realizará la evaluación de sus posturas. Esto para evitar la opacidad en la recopilación y el análisis de datos, desde una perspectiva ética. Así mismo, los datos de lenguaje corporal deben almacenarse y procesarse de manera segura para evitar brechas de seguridad y accesos no autorizados, particularmente en caso de que los datos incluyen información biométrica (Domínguez et al., 2021). Los desarrolladores de sistemas de evaluación de posturas deben asumir la responsabilidad de garantizar que sus aplicaciones sean éticas y respetuosas de la privacidad. Esto incluye la revisión constante de políticas y prácticas.

2.6. Evaluación de la efectividad de la retroalimentación

La retroalimentación desempeña un papel crucial en la evaluación automática de posturas en presentaciones orales, ya que proporciona a los presentadores información valiosa sobre su desempeño y les ayuda a mejorar. Evaluar la efectividad de la retroalimentación es esencial para garantizar que cumple su propósito de manera óptima.

En este sentido, la retroalimentación debe tener objetivos claros y específicos (Schneider et al., 2015). Antes de proporcionar retroalimentación, es esencial definir qué aspectos del lenguaje corporal se evaluarán y en qué áreas se espera que mejoren los presentadores. Por

otra parte, se deben establecer criterios de evaluación sólidos y relevantes. Esto implica definir qué se considera "positivo" y "negativo" en términos de posturas y gestos en el contexto de la presentación (Chen et al., 2014). Debe considerarse también la diversidad cultural y contextual en la interpretación de posturas. Lo que se considera "positivo" o "negativo" puede variar según la cultura, y los sistemas deben ser sensibles a estas diferencias (Schneider et al., 2015), es importante evaluar y mitigar cualquier sesgo que pueda surgir en la detección y clasificación de posturas (Carney et al., 2010).

En muchos casos puede ser útil también la comparación con el desempeño previo: la retroalimentación efectiva a menudo implica comparar el desempeño actual con el desempeño anterior. Esto permite a los presentadores ver su progreso y comprender en qué áreas han mejorado y en cuáles necesitan trabajar más (Wörtwein et al., 2015). También debe tenerse en cuenta que la retroalimentación debe basarse en datos objetivos y cuantificables. Por ejemplo, en lugar de decir "tu postura no fue buena", la retroalimentación podría ser "mantuviste una postura inclinada durante el 30% de la presentación" (Behoora y Tucker, 2015).

No debe olvidarse que la retroalimentación debe ser constructiva, alentadora y no perjudicial. Los comentarios negativos o destructivos pueden tener un impacto emocional en los presentadores y deben evitarse (Carney et al., 2010). Se deben resaltar los aspectos positivos del desempeño del presentador y, al señalar áreas de mejora, proporcionar sugerencias concretas y prácticas para el cambio (Holland et al., 2017). Además, la retroalimentación en tiempo real puede ser particularmente efectiva, ya que permite a los presentadores ajustar su lenguaje corporal de inmediato (Schneider et al., 2015).

Finalmente, los presentadores deben tener la oportunidad de evaluar la retroalimentación que reciben. Esto puede proporcionar información valiosa sobre la utilidad y efectividad de la retroalimentación proporcionada (Domínguez et al., 2021). Los sistemas de evaluación automática pueden incorporar adaptaciones en la retroalimentación en función de las necesidades y preferencias de los presentadores. Esto asegura que esta sea personalizada y relevante (Wörtwein et al., 2015).

2.7. Aplicaciones prácticas

La evaluación automática de posturas en presentaciones orales mediante el uso de algoritmos de detección de lenguaje corporal tiene una amplia gama de aplicaciones prácticas en diversos campos. A continuación, se detallan algunas de las aplicaciones más relevantes:

1. *Entrenamiento en Oratoria:* Una de las aplicaciones más directas es el entrenamiento en oratoria. Los sistemas de evaluación automática pueden proporcionar retroalimentación en tiempo real a los oradores en formación. Esto les ayuda a mejorar su lenguaje corporal y presentación en general (Schneider et al., 2015).

2. *Evaluación de Entrevistas de Trabajo:* Las entrevistas de trabajo son momentos críticos para evaluar la comunicación verbal y no verbal de los candidatos. La evaluación automática de posturas puede ayudar a los entrevistadores a identificar rasgos de lenguaje corporal relevantes para la toma de decisiones de contratación (Chen et al., 2014).

3. *Entrenamiento de Venta y Negociación:* Los profesionales de ventas y negociadores pueden beneficiarse al recibir retroalimentación sobre su lenguaje corporal. Esto puede ayudar a mejorar la persuasión y la influencia en situaciones de venta y negociación (Behoora & Tucker, 2015).

4. *Educación:* En entornos educativos, la evaluación automática de posturas puede ser utilizada para enseñar habilidades de presentación. Los estudiantes pueden practicar y recibir retroalimentación sin la necesidad de la presencia constante de un instructor (Wörtwein et al., 2015). Por otra parte, el sistema RAP (Retroalimentación Automática de Presentaciones), está diseñado para proporcionar retroalimentación automática sobre habilidades básicas de presentación oral a estudiantes de nivel inicial. Su propósito es ayudar a los estudiantes a mejorar sus habilidades de comunicación mediante el uso de análisis multimodal y sensores de bajo costo (Ochoa et at., 2018).

5. *Coaching Personal:* Los profesionales del *coaching* personal pueden utilizar estas herramientas para ayudar a sus clientes a mejorar sus habilidades de comunicación. La retroalimentación basada en datos objetivos es valiosa en el proceso de desarrollo personal (Carney et al, 2010).

6. *Evaluación de Políticas Públicas:* En el ámbito de la política y la toma de decisiones, la evaluación automática de posturas puede utilizarse para analizar y comparar el lenguaje corporal de los políticos durante discursos y debates. Esto puede proporcionar información sobre la autenticidad y la efectividad de su comunicación (Domínguez et al., 2021).

7. *Investigación en Comportamiento Humano:* Los investigadores pueden utilizar estos sistemas para analizar el comportamiento humano en contextos sociales. Esto es relevante para estudios en psicología, sociología y comunicación (Mehrabian, 1981).

8. *Desarrollo de Habilidades Sociales:* Las personas que buscan mejorar sus habilidades sociales, ya sea en presentaciones públicas, entrevistas o interacciones cotidianas, pueden utilizar la evaluación automática de posturas como una herramienta de autoevaluación y mejora continua (Schneider et al., 2015).

9. *Evaluación de Entrenadores y Profesores:* En entornos educativos y deportivos, los entrenadores y profesores pueden ser evaluados en su capacidad para comunicar instrucciones y liderar grupos utilizando la retroalimentación sobre su lenguaje corporal (Schneider et al., 2015).

10. *Terapia y Rehabilitación:* En el campo de la terapia y la rehabilitación, la evaluación de posturas puede ayudar a los terapeutas a trabajar con pacientes que necesitan mejorar su comunicación no verbal, como aquellos con trastornos del espectro autista (Shu, Wang & Zhan, 2020).

2.8. Evaluación de la calidad de la presentación oral con base en la detección automática de posturas

La calidad de una presentación oral basada en el número de posturas positivas detectadas por el proceso de detección de posturas con *MediaPipe* (Lugaresi et al., 2019) puede evaluarse de la siguiente manera: en primer lugar, y como ya se ha mencionado, debe definirse claramente cuáles son las posturas que se consideran "positivas" y "negativas" en el contexto de la presentación. Esto puede implicar identificar gestos corporales, posturas o movimientos específicos que se consideren deseables o indeseables para la audiencia o propósito. Entre los criterios más utilizados para evaluar una pose en presentaciones orales, se encuentran los siguientes (Laborda, 2019):

Aspectos positivos:

1. *Proyección de Confianza*: Una postura que demuestra seguridad y dominio del tema incluye tener la espalda recta y los hombros hacia atrás, mantener la cabeza en alto y la mirada firme para establecer contacto visual con la audiencia, y usar gestos abiertos y naturales que refuercen el mensaje verbal y generen confianza.

2. *Claridad y Expresividad*: La articulación precisa y el volumen adecuado permiten que el mensaje se escuche con claridad y se entienda. Un tono de voz variado y un ritmo adecuado mantienen la atención del público. Las expresiones faciales y los movimientos corporales congruentes con el mensaje verbal generan interés.

3. *Conexión con la Audiencia*: Un movimiento fluido y natural no distrae al público y permite una mejor comunicación. Orientarse hacia todo el público por igual y usar el espacio de manera adecuada, desplazándose con seguridad y sin invadir el espacio del público, también son importantes.

Aspectos negativos:

1. *Falta de confianza:* Una espalda encorvada o hombros hundidos, una cabeza baja o mirada evasiva, y gestos cerrados o nerviosos pueden demostrar inseguridad y falta de dominio del tema, generar desconfianza y distraer al público.

2. *Falta de claridad y expresividad*: Una articulación deficiente o un volumen bajo pueden dificultar la comprensión del mensaje. Un tono de voz monótono o un ritmo acelerado pueden aburrir al público y dificultar la atención. Las expresiones faciales y los movimientos corporales incongruentes con el mensaje verbal pueden confundir al público y restar credibilidad al mensaje.

3. *Desconexión con la audiencia*: Un movimiento rígido puede distraer al público y dificultar la comunicación. Orientarse hacia un solo punto o grupo puede hacer sentir excluidos a algunos presentes. Un uso inadecuado del espacio puede ser perjudicial, como por ejemplo invadiendo el espacio del público.

Otros aspectos a tener en consideración son (Laborda, 2019):

1. *Contexto de la presentación*: Adaptar la postura al tipo de público, la formalidad del evento y el tema de la presentación.

2. *Cultura y tradiciones*: Tener en cuenta las normas culturales que pueden afectar la percepción de la postura.

3. *Características Físicas del Orador*: Adaptar la postura a la altura, peso y condición física del orador.

3. Tecnologías usadas para la detección de posturas

En este capítulo se describen las tecnologías más empleadas para la detección de posturas, y a partir de este estudio se ha escogido la tecnología más adecuada para emplearse en el presente proyecto.

3.1. Tecnologías de detección de lenguaje corporal

Del análisis de la teoría de la persuasión y la efectividad de la presentación se puede comprender cómo los elementos no verbales, como el lenguaje corporal, pueden influir en la persuasión y la efectividad de una presentación oral. Dado que ciertas posturas y gestos pueden impactar en la percepción del público y en la persuasión del mensaje, se puede apreciar la relevancia de implementar tecnologías que permitan analizar y evaluar estos aspectos. La transición de la teoría a la aplicación tecnológica permite avanzar hacia un enfoque práctico para mejorar la calidad y el impacto de las presentaciones orales.

Para la revisión de las tecnologías de detección, se utilizó la técnica de *snowballing*, que implica comenzar con una referencia inicial y luego expandir el listado de fuentes a través de la identificación de nuevas fuentes a partir de las referencias bibliográficas de la fuente inicial y de las fuentes ya identificadas. El "*snowballing*" es un método de revisión bibliográfica que consiste en utilizar las referencias de un artículo relevante para identificar otros estudios relacionados. Este proceso se realiza de manera iterativa: primero, se selecciona un artículo base, luego se revisan sus referencias para encontrar otros trabajos relevantes, y así sucesivamente, expandiendo progresivamente la lista de artículos revisados. Este enfoque es útil para asegurar que se cubren de manera exhaustiva los estudios más pertinentes en un área de investigación (Kotsiantis et al., 2007).

En este caso, se inició con el trabajo de (Chen et al., 2014), seleccionando aquellas que parecen relevantes para el tema de la evaluación automatizada de habilidades de oratoria. En este trabajo, se ha considerado la investigación de (Chen et al., 2014) debido a que proporciona una base sólida para el análisis de habilidades de presentación utilizando señales multimodales. Aunque el estudio data de 2014, su relevancia radica en el enfoque pionero

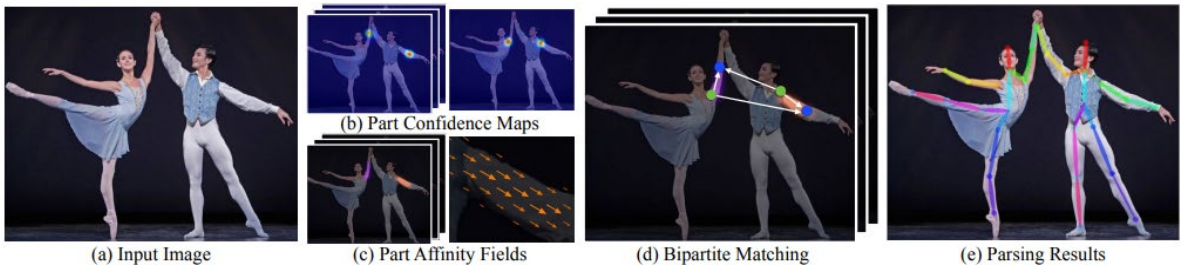
hacia la evaluación automatizada de habilidades de oratoria, sentando las bases para investigaciones posteriores (véase también las más de140 citas a este artículo en *Google Scholar*). No obstante, para asegurar que el trabajo presentado también incorpora las tecnologías y enfoques más recientes, se ha realizado una revisión exhaustiva de la literatura posterior y se han incorporado tecnologías actuales como *MediaPipe* y *OpenCV* que no solo optimizan la detección y análisis del lenguaje corporal en tiempo real, sino que también mejoran la accesibilidad y escalabilidad del sistema propuesto.

Este enfoque híbrido permite aprovechar las contribuciones fundamentales de trabajos previos como el de (Chen et al., 2014), al mismo tiempo que se integran las innovaciones más recientes en el campo, garantizando que el sistema desarrollado esté alineado con los avances tecnológicos actuales. Una vez identificadas estas nuevas fuentes, se procedió a revisar sus propias referencias bibliográficas, lo que permitió encontrar aún más fuentes relacionadas. Las tecnologías de detección de lenguaje corporal desempeñan un papel esencial en el desarrollo de sistemas de evaluación automática de posturas en presentaciones orales. Estas tecnologías permiten la captura y análisis de señales no verbales, como gestos y posturas, que son fundamentales para comprender la comunicación no verbal en el entorno de las presentaciones orales. A continuación, se enumeran unas cuantas tecnologías utilizadas para tal fin:

1. *OpenPose*: *OpenPose* es una tecnología de visión por computadora que se ha vuelto fundamental en la detección de puntos de articulación en el cuerpo humano. Utiliza una red neuronal de convolución (CNN) para rastrear y etiquetar las articulaciones en imágenes o videos. Esta tecnología puede ser utilizada en aplicaciones de seguimiento de lenguaje corporal y proporcionar información detallada sobre la postura y los gestos de un presentador en tiempo real (Cao et al., 2019). La secuencia general de trabajo de esta tecnología se presenta en la ***Figura 4*.**

Figura 4

Secuencia general de trabajo de OpenPose (Cao et al., 2019)



Fuente: Cao et al. (2019). *Secuencia general de trabajo de OpenPose.* (a) Se toma la imagen completa como entrada para que la CNN prediga conjuntamente: (b) mapas de confianza para la detección de partes del cuerpo y (c) campos de afinidad (PAF) para la asociación de partes. (d) El paso de análisis realiza un conjunto de coincidencias bipartitas para estimar y asociar las partes del cuerpo. (e) Finalmente, se ensamblan las posturas de cuerpo completo para todas las personas en la imagen.

2. *Kinect de Microsoft*: El sensor *Kinect* es un dispositivo de detección de movimiento que se ha utilizado en investigaciones y aplicaciones de evaluación de lenguaje corporal. Utiliza una combinación de cámaras RGB y sensores de profundidad para capturar y seguir el movimiento del cuerpo en 3D. Esto permite una evaluación precisa de la postura y los gestos de un orador durante una presentación (Yang, 2014; Murata & Shin, 2014; Del Río y& Martín-Gutiérrez, 2020).

3. *Tecnología de seguimiento de ojos*: La tecnología de seguimiento de ojos, como los rastreadores de mirada (*eye trackers*), puede ser utilizada para analizar la dirección de la mirada de un presentador y la audiencia durante una presentación. Esto puede revelar qué elementos visuales atraen más la atención y cómo la atención se distribuye en la sala (Richardson & Spivey, 2004).

4. *Sensores de movimiento corporal*: Se han desarrollado dispositivos portátiles y sensores de movimiento que pueden ser utilizados para evaluar la postura y los gestos de un presentador. Estos dispositivos pueden incluir acelerómetros y giroscopios para capturar el movimiento y la orientación del cuerpo (Pons-Moll et al., 2010).

5. *Tecnología de reconocimiento facial*: El reconocimiento facial se centra en las expresiones faciales y se considera relevante en el contexto de la comunicación no verbal. El análisis de expresiones faciales puede proporcionar información valiosa sobre las emociones y el estado emocional del presentador (Baron, 1981; Kaur et al., 2020).

6. *Aprendizaje automático y visión por computadora*: Estas tecnologías son fundamentales para el procesamiento y análisis de datos de lenguaje corporal capturados por las herramientas mencionadas anteriormente. Los algoritmos de aprendizaje automático, como las redes neuronales, se utilizan para clasificar gestos, posturas y expresiones faciales en categorías relevantes, como "positivas" o "negativas." (Behoora & Tucker, 2015).

7. *Realidad virtual y aumentada*: Estas tecnologías permiten la creación de entornos interactivos de presentación donde los gestos y las posturas de un presentador se pueden analizar y evaluar en un entorno virtual o aumentado. Esto puede ser valioso para la capacitación y la retroalimentación en un entorno simulado (Hsiao y& Rashvand, 2011).

3.2. Bibliotecas y herramientas utilizadas en *machine learning* y su aplicación en la clasificación de posturas

El *machine learning* es una rama de la inteligencia artificial que se ocupa de crear sistemas capaces de aprender de los datos y mejorar su rendimiento sin necesidad de una programación explícita (Behoora & Tucker, 2015). Para ello, se pueden utilizar una serie de bibliotecas y herramientas que facilitan el desarrollo y la implementación de algoritmos y modelos de aprendizaje. Entre las bibliotecas más populares y utilizadas en *machine learning* se encuentran:

- *TensorFlow*[2]*:* es una plataforma de código abierto para el lenguaje de programación *Python* desarrollada por *Google* que permite crear y entrenar redes neuronales profundas para diversas tareas, como visión artificial, procesamiento del lenguaje natural, reconocimiento de voz, etc. (Abadi et al., 2015). *TensorFlow* ofrece una interfaz de alto nivel llamada *Keras*[3], que simplifica la creación y el entrenamiento de los modelos (Abadi et al., 2015).

- *PyTorch*[4]*:* es otra plataforma de código abierto para el lenguaje de programación *Python* desarrollada por Facebook que también permite crear y entrenar redes neuronales profundas, con la ventaja de que ofrece un modo dinámico que facilita la depuración y la experimentación (Paszke et al., 2019). *PyTorch* también cuenta con una interfaz de alto nivel llamada *Torchvision* (Paszke et al., 2019) (TorchVision: sus mantenedores y contribuyentes, 2016), que proporciona modelos pre-entrenados y conjuntos de datos para visión artificial.

- *Scikit-learn:* es una biblioteca de aprendizaje automático de código abierto para el lenguaje de programación *Python* que ofrece una amplia gama de algoritmos y herramientas para *machine learning*, como clasificación, regresión, reducción de dimensionalidad, selección de características, etc. (Pedregosa et al., 2011). *Scikit-learn* es compatible con otras bibliotecas como *NumPy* (Pedregosa et al., 2011; Harris et al., 2020), *SciPy* y *Matplotlib* que permiten manipular y visualizar los datos (Carey et al., 2020; Hunter, 2007).

- *MediaPipe*[5]: es un *framework* de código abierto desarrollado por *Google* para el lenguaje de programación *Python* que permite crear aplicaciones de visión artificial en tiempo real, como detección y seguimiento de rostros, manos, posturas, objetos, etc. (Lugaresi et al., 2019). *MediaPipe*[4] ofrece una serie de soluciones predefinidas

[2] Software disponible en: https://www.tensorflow.org.
[3] Software disponible en: https://keras.io
[4] Software disponible en: https://pytorch.org
[5] Software disponible en: https://developers.google.com/mediapipe

que se pueden ejecutar en diferentes plataformas, como web, móvil o escritorio (Lugaresi et al., 2019).

Una de las aplicaciones más interesantes del *machine learning* es, precisamente, la clasificación de gestos y posturas, que consiste en identificar y reconocer las acciones o movimientos que realiza una persona a partir de imágenes o vídeos (Behoora & Tucker, 2015). Esta tarea tiene múltiples aplicaciones prácticas, como el control por gestos, la realidad aumentada, la rehabilitación física, el análisis deportivo, etc.

Como ya se ha dicho, para realizar la clasificación de gestos y posturas se pueden utilizar diferentes técnicas y modelos de *machine learning*, como redes neuronales convolucionales (CNN), redes neuronales recurrentes (RNN), redes neuronales gráficas (GCN), etc. Una de las herramientas más potentes y sencillas para esta tarea es *MediaPipe*[4] (Lugaresi et al., 2019), la cual ofrece una solución predefinida llamada *Pose Estimation* (Kim et al., 2023), que permite detectar y estimar las posturas de una o varias personas en tiempo real.

3.3. Selección de tecnología

Luego de evaluar la diferentes tecnologías y plataformas, se ha optado por utilizar *Pose Estimation* de *MediaPipe*[4] (Lugaresi et al., 2019; Kim et al., 2023), lo cual se justifica por varias razones fundamentales. En primer lugar, la solución predefinida que ofrece *MediaPipe*[4] (Lugaresi et al., 2019) simplifica considerablemente el proceso de implementación, ya que proporciona una herramienta lista para usar que puede ser integrada fácilmente en aplicaciones de detección de gestos y posturas. Esto ahorra tiempo y recursos que de otro modo se emplearían en desarrollar y entrenar un modelo propio desde cero. Además, el hecho de que *MediaPipe*[6] esté disponible en *Python*, un lenguaje de programación popular en el ámbito del *machine learning*, ofrece importantes ventajas en términos de accesibilidad y flexibilidad. *Python* cuenta con una amplia gama de bibliotecas y *frameworks* para *machine learning*, lo que facilita la integración de *Pose Estimation* de *MediaPipe*[5] (Kim et al., 2023) con otras herramientas y técnicas de *machine learning* que puedan ser necesarias para la clasificación y evaluación de gestos y posturas. Esta combinación de una solución predefinida y la compatibilidad con *Python* permite una

[6] Software disponible en: https://developers.google.com/mediapipe

implementación eficiente y efectiva del sistema de clasificación de gestos y posturas, maximizando así el potencial de éxito del proyecto.

La solución *Pose Estimation* de *MediaPipe*[5] utiliza un modelo basado en CNN llamado *BlazePose*, que consta de dos partes: un detector y un estimador. El detector se encarga de localizar a las personas en la imagen o el vídeo y generar un recuadro alrededor de cada una. El estimador se encarga de analizar el recuadro generado por el detector y estimar las coordenadas 2D o 3D de los puntos clave del cuerpo humano, como la cabeza, los hombros, los codos, las manos, las rodillas, los pies, etc. De esta forma, se obtiene una representación vectorial de la pose o postura de cada persona.

Es importante señalar que se utilizó *MediaPipe* para este proyecto debido a su capacidad para proporcionar un seguimiento de posturas y gestos corporales con alta precisión, todo ello utilizando tecnologías accesibles y de código abierto. Sin embargo, también se consideraron otras alternativas populares, como *OpenPose* y el uso de sensores de movimiento como *Kinect*.

Al respecto, *OpenPose*, es una herramienta muy potente para la detección de puntos clave del cuerpo humano. Sin embargo, *OpenPose* requiere mayores recursos computacionales, lo que podría limitar su uso en entornos con menos capacidad de procesamiento, como dispositivos móviles. Además, *OpenPose* tiene una curva de aprendizaje más pronunciada y no se integra tan fácilmente con otras herramientas de *Python* como *MediaPipe*.

Por otra parte, aunque *Kinect* ofrece una excelente precisión en la captura de movimientos en 3D, su uso está limitado por la necesidad de hardware especializado, lo que reduce su accesibilidad. Además, el desarrollo de nuevas aplicaciones utilizando *Kinect* puede ser costoso y requiere un entorno específico de desarrollo que no es tan flexible como *Python*.

Entonces, *MediaPipe* fue elegida por su equilibrio entre precisión, accesibilidad y facilidad de integración con *Python*. A diferencia de otras alternativas, *MediaPipe* permite el desarrollo de soluciones portátiles que pueden ser fácilmente escalables y aplicadas en una variedad de entornos sin la necesidad de hardware especializado. Esto no solo mejora la accesibilidad del sistema desarrollado, sino que también asegura que el sistema puede ser

utilizado en dispositivos con capacidades computacionales limitadas, como *smartphones* y *laptops.*

A partir de esta representación vectorial se puede realizar la clasificación de gestos y posturas utilizando diferentes algoritmos o modelos de *machine learning*, los cuales se pueden entrenar con conjuntos de datos etiquetados que contengan ejemplos de diferentes gestos o posturas a reconocer.

4. Implementación

En este capítulo se describe el procedimiento seguido durante el desarrollo del proyecto, cuyo objetivo fue desarrollar un sistema de reconocimiento y evaluación del lenguaje corporal en presentaciones orales mediante técnicas de *machine learning*. Primero, se explica por qué el análisis de las bibliotecas y herramientas utilizadas es fundamental para la construcción del sistema. Luego, se detallan los pasos realizados para crear, entrenar y evaluar el modelo.

4.1. Aprendizaje automático y redes neuronales de convolución (CNN) en la evaluación de posturas en presentaciones orales

El análisis de las bibliotecas y herramientas utilizadas en *machine learning* (ML) y su aplicación en la clasificación de gestos y posturas sienta las bases para adentrarnos en el uso específico de técnicas de aprendizaje automático y redes neuronales de convolución (CNN) en la evaluación de posturas en presentaciones orales.

El aprendizaje automático, y en particular las redes neuronales de convolución desempeñan un papel crucial en el desarrollo de algoritmos (IBM, 2020) los cuales pueden emplearse en la evaluación automática de posturas en presentaciones orales. Estas tecnologías permiten el procesamiento, análisis y clasificación de datos de lenguaje corporal de manera eficiente y precisa.

1. *Extracción de características*: Las redes neuronales de convolución son especialmente adecuadas para extraer características significativas de datos de imágenes (IBM, 2020), como los generados por tecnologías de detección de lenguaje corporal. Al igual que el cerebro humano descompone la información en diferentes características, como bordes, formas y texturas, antes de unirlas para entender la imagen en su conjunto, las CNN hacen algo similar. Esto permite a las CNN identificar y aprender patrones complejos en los datos de imágenes, lo que las hace especialmente adecuadas para tareas como la detección del lenguaje corporal. En el contexto de la evaluación de posturas, las CNN pueden identificar patrones y

características clave en los gestos y las posturas, como la posición de las articulaciones y la orientación del cuerpo.

2. *Entrenamiento de modelos*: Para evaluar y clasificar posturas como "positivas" o "negativas", es necesario entrenar un modelo de aprendizaje automático. Este proceso implica proporcionar al modelo un conjunto de datos etiquetado que contenga ejemplos de gestos y posturas clasificados. El modelo aprende a partir de estos ejemplos y desarrolla la capacidad de reconocer patrones y relaciones en los datos.

3. *Selección de características relevantes*: Las CNN pueden ser utilizadas para seleccionar las características más relevantes del lenguaje corporal que están relacionadas con la efectividad de una presentación. Esto incluye gestos específicos, como movimientos de las manos, la orientación del cuerpo, la expresión facial y otros indicadores no verbales que contribuyen a la percepción de la audiencia.

4. *Clasificación de gestos y posturas*: Una vez que el modelo está entrenado y las características relevantes se han extraído, el siguiente paso es clasificar posturas en categorías, como "positivas" o "negativas". Esto implica la aplicación de algoritmos de clasificación, que asignan etiquetas a los datos de entrada en función de las características identificadas (Gulati, 2022).

5. *Evaluación continua*: Los modelos de aprendizaje automático pueden utilizarse para evaluar el lenguaje corporal en tiempo real durante una presentación (Gulati, 2022). Esto permite el seguimiento de la efectividad de las posturas y la generación de retroalimentación instantánea.

6. *Mejora con la experiencia*: A medida que se recopila más información sobre presentaciones y retroalimentación proporcionada, los modelos de aprendizaje automático pueden mejorar su precisión y capacidad de evaluación (IBM, 2020). El aprendizaje continuo y la adaptación a diferentes estilos de presentación son aspectos clave de estos sistemas.

7. *Transparencia y explicabilidad*: Es importante que los modelos de aprendizaje automático utilizados en la evaluación de posturas en presentaciones orales sean transparentes y explicables (IBM, 2020). Esto permite que los presentadores comprendan por qué se han dado ciertas calificaciones y retroalimentación, lo que es fundamental para la mejora de sus habilidades.

4.2. Detección y segmentación de posturas con *MediaPipe*

La detección y segmentación de posturas con *MediaPipe* (Lugaresi et al., 2019) es un proceso que se basa en la capacidad de esta biblioteca de visión por computadora para detectar puntos clave en el cuerpo y seguir la postura de una persona en tiempo real. A continuación, se describe el proceso de detección y segmentación de posturas con este software de manera general:

1. Una vez que se ha instalado *MediaPipe* y cargado las bibliotecas correspondientes en el entorno de desarrollo, debe inicializarse *MediaPipe* para utilizar sus modelos de detección de posturas.
2. Se abre la cámara o se carga un video, que se desee analizar. Para esto se puede utilizar la librería *OpenCV* (Itseez, 2015) (cv2) para realizar la captura de video.
3. El proceso principal implica el procesamiento de cada fotograma del video. Debe capturarse cada fotograma, convertirlo a un espacio de color uniforme (RGB, escala de grises) y luego utilizar el modelo de *MediaPipe* para detectar la pose en ese fotograma.
4. Una vez que se ha detectado la pose en el fotograma, se puede dibujar la postura identificada en el mismo. *MediaPipe* proporciona herramientas para dibujar los puntos clave y las conexiones entre ellos en el fotograma. Finalmente, se puede mostrar el resultado visualizado en una ventana.

El proceso descrito anteriormente es un flujo básico para la detección y segmentación de posturas con *MediaPipe.* La biblioteca se encarga de detectar puntos clave en el cuerpo, lo que permite identificar la pose. A continuación, se describe detalladamente el proceso que se

ha llevado a cabo para adaptar el sistema a las necesidades y aplicaciones específicas, que requiere el presente proyecto.

4.3. Implementación del sistema para clasificación y evaluación del lenguaje corporal

En adelante se explicará detalladamente la construcción del sistema, desde la recolección de la información hasta la interfaz desarrollada para probar el sistema de clasificación y evaluación del lenguaje corporal.

4.3.1. Recolección de datos

Se dispone de un grupo de 26 voluntarios, compuesto por 13 hombres y 13 mujeres, con edades entre 22 y 35 años, estudiantes y profesionales universitarios de diferentes nacionalidades entre las cuales se incluyen: española, chilena, colombiana y ecuatoriana. Este rango de edad se eligió debido a que representa una etapa en la que las habilidades de presentación oral son altamente valoradas y requeridas tanto en el ámbito académico como profesional, permitiendo así una evaluación precisa de las competencias comunicativas y del lenguaje corporal en contextos que reflejan situaciones reales de presentaciones.

El procedimiento de reclutamiento se inició con la difusión de la convocatoria a través de diversos canales, incluyendo anuncios en las plataformas internas de la universidad, correos electrónicos a estudiantes de cursos avanzados y redes de contacto profesional dentro del entorno académico. Los datos demográficos de los participantes mostraron una diversidad en términos de formación académica y experiencia profesional, incluyendo estudiantes de diversas disciplinas como ciencias sociales, ingeniería y artes, así como jóvenes profesionales con experiencia en campos como la consultoría, la docencia, y el emprendimiento. Esta variedad proporcionó una base de datos para analizar el lenguaje corporal en un espectro amplio de contextos de presentación, permitiendo a su vez extrapolar los resultados del estudio a diferentes escenarios profesionales y académicos.

A cada participante se le indica que realice una pose específica, y se le toman cuatro fotografías por cada pose: de frente, de espaldas, de perfil derecho e izquierdo. El conjunto de datos consta de un total de 1040 imágenes, esto incluye las posturas que se han

mencionado en la ***Tabla 1***. Este repositorio de imágenes se puede encontrar en la carpeta de *Google drive*[7].

Para el proceso de etiquetado de las imágenes se le ha pedido a un entrenador en oratoria que realizase el trabajo, que consistió en clasificar el conjunto de datos de entrenamiento en posturas positivas y posturas negativas, por cada participante. Este entrenador posee una formación sólida en técnicas de expresión oral y lenguaje corporal, habiendo trabajado extensamente en la capacitación de estudiantes y profesionales para mejorar sus habilidades de presentación y comunicación efectiva. Su experiencia abarca tanto el ámbito académico como el empresarial, lo que le permite comprender las diversas demandas comunicativas que pueden surgir en diferentes contextos.

El conjunto de datos se etiqueta manualmente en función de los criterios derivados de la literatura y presentados en la ***Tabla 1***, asignando a cada imagen una etiqueta de "positiva" o "negativa". Estas etiquetas se utilizarán como las clases que el modelo debe aprender a predecir. A continuación, se presenta la figura 5 que muestra ejemplos de captura del conjunto de datos mediante la cámara de un dispositivo móvil. Estas imágenes son realistas y se toman con la mayor calidad posible.

[7] https://bit.ly/3XLnmh7

Figura 5

Ejemplos de captura del conjunto de datos mediante la cámara de un dispositivo móvil

Fuente: Elaboración propia.

4.3.2. Preprocesamiento de las imágenes

Una vez que las imágenes están etiquetadas, se realiza el preprocesamiento para prepararlas para el entrenamiento. Para este proyecto se llevaron a cabo los siguientes pasos:

4.3.2.1. Redimensionamiento

Las imágenes se redimensionan a un tamaño uniforme para que puedan ser procesadas por el modelo. Se proporciona al sistema la imagen de entrada. El sistema tomará la imagen, la convertirá al tamaño de (224,224) i.e., altura y ancho.

La resolución elegida para el análisis de las imágenes y videos en este proyecto fue de X píxeles por Y píxeles (224,224). Este tamaño de entrada es escogido por ser el más utilizado en muchas arquitecturas de redes neuronales convolucionales (CNN), (i) es compatible con redes preentrenadas en grandes conjuntos de datos (ii) equilibrio entre la calidad de la imagen y la eficiencia computacional, lo que quiere decir que este tamaño es lo suficientemente pequeño para realizar procesos computacionales eficientes pero lo suficientemente grande para capturar información visual relevante (iii) simplifica el preprocesamiento de las imágenes, ya que asegura que todas las imágenes tengan la misma resolución y tamaño, lo que es necesario para alimentar consistentemente a la red (iv) es el tamaño estándar en

conjunto de datos populares, por ello es practico escoger este tamaño ya que facilitaría un posible proceso de comparar diferentes arquitecturas y enfoques de manera justa y estandarizada.

Cabe mencionar que, la resolución se seleccionó también para asegurar la compatibilidad con los dispositivos de captura de imágenes utilizados en el estudio, como cámaras de *smartphones* y *webcams* estándar, que suelen operar en esta resolución de manera predeterminada. Esto facilita la recolección de datos y asegura que el sistema sea aplicable en un entorno realista y accesible.

Además, se revisó estudios previos que emplean técnicas de visión por computadora para la detección de posturas, y se observó que resoluciones similares han sido efectivas en aplicaciones comparables, como se evidencia en trabajos de investigación recientes. Esta resolución, por tanto, no es arbitraria, sino que se seleccionó cuidadosamente para optimizar la precisión del modelo y su aplicabilidad en condiciones reales.

4.3.2.2. Utilización de la biblioteca *MediaPipe*

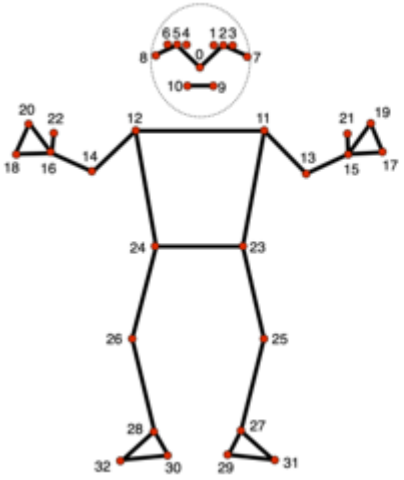
El sistema utiliza la biblioteca *MediaPipe,* que emplea dos modelos distintos para identificar y analizar posturas humanas dentro de imágenes. El proceso comienza con el primer modelo, cuya función es reconocer si hay personas en la imagen y, de ser así, determinar ciertos puntos clave de su postura. Estos puntos clave son esenciales porque actúan como una guía inicial para entender la pose general del cuerpo. Una vez que se ha confirmado la presencia de personas y se han identificado los puntos clave, entra en acción el segundo modelo.

Este segundo modelo tiene una tarea más detallada: se encarga de mapear la postura completa de la persona, proporcionando una imagen más precisa y exhaustiva de la pose mediante la estimación de 33 puntos de referencia por defecto. Estos puntos no son solo bidimensionales; ofrecen una perspectiva tridimensional, lo que significa que pueden mostrar la profundidad y el contorno completo del cuerpo en el espacio. La ***Figura 6***, a la que se hace referencia en adelante, ilustra este mapeo tridimensional, mostrando cómo el modelo de referencia de pose puede crear una representación detallada y volumétrica de la postura humana. En resumen,

MediaPipe detecta personas y puntos clave de la postura y luego los utiliza para construir un modelo tridimensional completo de la pose humana.

Figura 6

Modelo de landmarks para detección de posturas utilizado por MediaPipe



Fuente: Elaboración propia

En el presente proyecto se utiliza *MediaPipe* para extraer y visualizar puntos clave de posturas a partir de imágenes y generar sus puntos clave (*landmarks*) como se muestra en la ***Figura** 7*.

Figura 7

Ejemplos de landmarks generados mediante la biblioteca MediaPipe

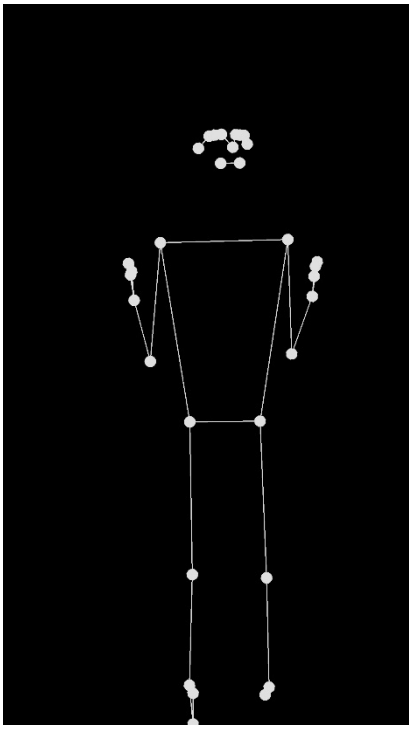
Fuente: Elaboración propia.

4.3.2.3. Generación del lienzo negro

En el siguiente paso, se genera una imagen negra aleatoria, es decir, todos los valores de píxeles son 0, luego los puntos clave generados anteriormente se dibujan en esa imagen, como se muestra en ***Figura 8.*** La razón de utilizar un lienzo negro es permitir que el modelo comprenda las diferentes posturas relacionadas con respecto a cada clase.

Figura 8

Ejemplo de landmarks dibujados *en el fondo negro*



Fuente: Elaboración propia.

4.3.2.4. Normalización

En la implementación del sistema, se decidió utilizar una resolución específica para las imágenes debido a varios factores técnicos clave. Primero, el proceso de normalización de los valores de píxeles a un rango entre 0 y 1, dividiendo todos los valores de píxeles por su valor más alto, que es 255, es fundamental para mejorar la eficiencia del entrenamiento del modelo, ya que garantiza que el modelo se enfoque en las características más relevantes de las posturas sin la influencia de ruido visual de alta resolución.

Esta elección no es arbitraria; está basada en la necesidad de mantener un balance entre la calidad de la información que el modelo recibe y la eficiencia computacional del sistema. Resoluciones más altas podrían aumentar la carga computacional sin necesariamente

proporcionar mejoras significativas en la precisión del modelo, especialmente considerando que el objetivo es la detección y clasificación de posturas.

4.3.2.5. Aumento de datos

La técnica de aumento de datos (*data augmentation)* se trata de diferentes métodos utilizados en *machine learning*, en diferentes contextos con la finalidad de incrementar de una manera artificial el tamaño y la diversidad de un conjunto de datos de entrenamiento. Este método consiste en la creación de nuevas muestras partiendo del conjunto original y empleando diferentes transformaciones como rotaciones, desplazamientos en el eje vertical y horizontales, y otros ajustes que modifican ligeramente las imágenes originales sin alterar la etiqueta asociada a ellas (Shorten, & Khoshgoftaar, 2019). Esta técnica se utilizó en este proyecto, principalmente porque el conjunto de datos disponible era limitado, y con esta técnica se mejora la capacidad de generalización del modelo al exponerlo a variaciones más amplias de las imágenes durante el entrenamiento. En la ***Figura 9*** se observan las acciones realizadas para el aumento de datos usando la clase *ImageDataGenerator*[8]. En el ***anexo B***, se encuentra el código completo que se ha utilizado.

Figura 9

Generador de imágenes usando la clase *ImageDataGenerator*

```
def augment_images_in_folder(input_folder, output_folder, num_augmented_images):
    # Create an instance of the ImageDataGenerator with augmentation parameters
    datagen = ImageDataGenerator(
        rotation_range=40,       #Random rotation between 0-40 degrees
        width_shift_range=0.2,   #Random horizontal shift
        height_shift_range=0.2,  #Random vertical shit
        shear_range=0.2,         #shearing transformation
        zoom_range=0.2,          #Random zoom
        horizontal_flip=True,    #Ramdon horizontal flipping
        fill_mode='nearest'      #Fill in the pixels after augmentation
    )
```

Fuente: Elaboración propia.

[8] https://www.tensorflow.org/api_docs/python/tf/keras/preprocessing/image/ImageDataGenerator

Las acciones que se realizaron son (i) rotación aleatoria entre 0 y 40 grados, esto es especialmente útil ya que hace que el modelo sea más robusto frente a diferentes ángulos desde donde pueda estar el presentador (ii) desplazamientos horizontal y vertical, son útiles para enseñarle al modelo a predecir independientemente de la posición exacta del presentador (iii) transformación *shear,* puede ayudar al modelo a ser menos susceptible a diferentes perspectivas que pueda tener la imagen que ha sido captada desde diferentes ángulos de grabación (iv) zoom aleatorio, al aplicar este zoom se está ayudando al modelo a la capacidad de reconocer al presentador en diferentes tamaños y distancias (v) volteo horizontal, es útil para entrenar al modelo a que no dependa de una orientación especifica (vi) modo de relleno, es útil para evitar que se creen objetos no deseados en la imagen generada por la transformación, tienen el objetivo de aumentar la variabilidad de los datos de entrenamiento, sin necesidad de recopilarlos todos, esto ayuda a que el modelo pueda generalizar mejor (estas técnicas se describen en: https://bit.ly/4e1wVOF) según se describe en (Shorten, & Khoshgoftaar, 2019).

Estas características son aplicadas de manera aleatoria sobre cada imagen, no todas las características se aplican a cada imagen, se hace de una en una. Se escogió como parámetro de diseño generar 3000 imágenes, debido al costo computacional que conlleva generar estas imágenes y siendo esa una limitante en el desarrollo de este proyecto.

Al final el conjunto de datos empleado se ha dividido de la siguiente manera, conjunto de entrenamiento y conjunto de prueba, utilizando el 70% para entrenamiento y el 30% para pruebas, estos porcentajes son valores típicos por diferentes motivos, primero usar un 70% de datos para entrenamiento proporciona una cantidad razonablemente grande de datos para que el modelo logre aprender los patrones. Mientras más datos tenga el modelo para entrenar, mejor podrá ajustar sus parámetros y mejorar su capacidad para generalizar. El 30% elegido para el conjunto de prueba, permite un tamaño apropiado de datos no vistos por el modelo, esto es muy importante ya que permite realizar una evaluación robusta del rendimiento del modelo como se muestra en (Singh, & Misra, 2020). Sí el conjunto de datos de prueba es demasiado pequeño, la evaluación no se consideraría de alguna forma confiable. Después de realizar el proceso de aumento de datos el conjunto de datos final se muestra a continuación. En ***Figura 10*** se observa el tamaño de las carpetas que contienen el conjunto de datos de

entrenamiento y prueba. El total de las imágenes de conjunto de datos son 5197 imágenes el resultado que es el resultado final del aumento de datos más la suma de los datos originales.

Conjunto de entrenamiento:

Cantidad de imágenes positivas: 2116

Cantidad de imágenes negativas: 1671

Conjunto de prueba:

Cantidad de imágenes positivas: 747

Cantidad de imágenes negativas: 663

Figura 10

Conjunto de datos de entrenamiento y conjunto de prueba

Fuente: Elaboración propia

4.3.3. Entrenamiento del modelo

Para el entrenamiento del modelo se utilizó un *notebook*[9] de *Google Colab.* Se utilizó este entorno para entrenar el modelo dada la ejecución en la nube y acceso a GPU/TPU (*Graphics Processing Unit/ Tensor Processing Unit*), lo que acelera significativamente el entrenamiento de modelos de *deep learning,* así como la posibilidad de integrarse con *Google Drive* facilitando la gestión de los datos y resultados.

[9] https://colab.research.google.com

El siguiente paso es diseñar el modelo de *machine learning* que se utilizará para la clasificación. Para el desarrollo del modelo se llevaron a cabo los siguientes pasos, el código completo se encuentra en el repositorio de *Google drive*[10]

A continuación, se describen los pasos que se realizaron para entrenar y probar el modelo:

a. Se define la forma de entrada y se crean dos modelos base: *MobileNetV2*[11] y *DenseNet169*[12]. Este paso se refiere a que se ha cargado dos modelos que han sido entrenados previamente, *MobileNetV2 y DenseNet169*. Ambos modelos están pre-entrenados, lo que significa que ya han sido entrenados en un conjunto complejo de imágenes de más de 10 clases y han obtenido buenos resultados. Estos modelos definen la forma de entrada de las imágenes al modelo, es decir como todas las imágenes tienen una anchura y una altura constantes de 224 x 224, se debe establecer la forma de entrada de estos modelos en 224 x 224 para que puedan aceptar las imágenes.

b. Se congelan las capas en ambos modelos base para evitar su entrenamiento. Este paso busca que la arquitectura básica de ambos modelos no se vuelva a entrenar, como sus pesos ya están optimizados, restablecer estos pesos implicaría entrenar toda la arquitectura desde cero y puede llevar mucho tiempo.

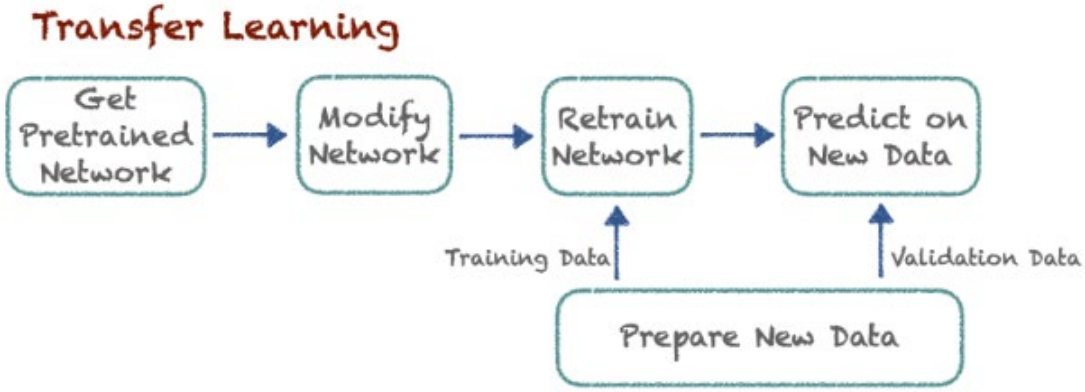
c. Se concatenan las salidas de los modelos base y se agregan capas *Dense* adicionales con *BatchNormalization*[13], *Dropout*[14] y funciones de activación. Esta parte del proceso lo que busca es aplicar básicamente la técnica de *transfer learning*, comúnmente usado en *machine learning* en el que un modelo que es entrenado previamente en una tarea es reutilizado o ajustado para una nueva tarea, valiéndose de los conocimientos que ha adquirido en el entrenamiento previo, lo que permite entrenar el nuevo modelo de una manera más ágil y con menos datos como se observa en el análisis presentado en Singh, V., & Misra, A. K. (2020).

   Por ejemplo, *DenseNet* está entrenado en 100 clases, lo que significa que puede predecir 100 clases diferentes, pero para el caso específico del proyecto se tienen únicamente 2 clases, positiva y negativa. Por lo tanto, se necesita cambiar estas capas con respecto a la necesidad particular del proyecto. En la ***Figura 11*** se observa el proceso mencionado:

[10] https://bit.ly/3XLnmh7
[11] https://pytorch.org/hub/pytorch_vision_mobilenet_v2/
[12] https://keras.io/api/applications/densenet/#densenet169-function
[13] https://keras.io/api/layers/normalization_layers/batch_normalization/
[14] https://keras.io/api/layers/regularization_layers/dropout/

Figura 11

Proceso de *transfer learning*



Fuente: https://blogs.mathworks.com/deep-learning/2023/06/11/transfer-learning-made-easy/

d. Se compila el modelo con el optimizador *Adam*[15] y la función de pérdida de entropía cruzada binaria[16]. Básicamente esta sección se refiere a agregar el algoritmo de Adam, en el proceso de entrenamiento del modelo, después de cada iteración el modelo verifica su rendimiento e intenta ajustar su valor de ponderación para generar respuestas más precisas. Para este proceso se utiliza un optimizador que en caso del presente proyecto es Adam, se ha escogido gracias a sus principales ventajas como son la adaptación de la tasa de aprendizaje, corrección de sesgo entre otros. Adam es robusto y eficiente, lo que lo hace ideal para este proyecto.

e. Se configuran *callbacks*[17] estos son usados para monitorear el entrenamiento del modelo en cada iteración, para el desarrollo del proyecto, se han *usado early stopping,* y *learning rate. Early stopping:* se encarga de comprobar si el rendimiento del modelo deja de mejorar, y si al cabo de 2 o 3 iteraciones consecutivas no mejora, entonces el entrenamiento se detendrá, es decir ayuda a prevenir el sobreajuste (*overfitting*) y a evitar tiempo de entrenamiento innecesario. *Learning rate*: este hiperparámetro define cuan grandes son los ajustes que el optimizador da al actualizar los pesos del modelo durante el proceso de aprendizaje. En este proyecto se utilizó una técnica muy recomendada para el entrenamiento de los modelos, y se trata de disminuir la tasa de aprendizaje (*learning rate*).

f. Se entrena el modelo en los datos de entrenamiento y se valida en los datos de prueba. Para el entrenamiento del modelo se han escogido las epócas (*epochs)*, el tamaño del lote (*batch)* como se muestra en la ***Figura 12***. Estos parámetros han sido escogidos bajo criterio propio, y se han ajustado de manera experimental, encontrando que estos parámetros son los que mejor resultado ha dado en proceso de entrenamiento.

[15] https://keras.io/api/optimizers/adam/
[16] https://keras.io/2.15/api/losses/probabilistic_losses/
[17] https://keras.io/api/callbacks/

Figura 12
Parámetros de entrenamiento del modelo

```
# Train the stacked model with the specified training parameters and callbacks
stacked_history = stacked_model.fit(
    train_generator,                    # Training data generator providing batches of images and labels for training
    epochs=100,                         # Number of epochs to train the model; each epoch involves one full pass through the training data
    batch_size=15,                      # Batch size to be used during training; determines the number of samples processed before the model's weights are updated
    validation_data=test_generator,   # Validation data generator providing batches of images and labels for validation
)
stacked_model.save('pose_model.h5')
```

Fuente: Elaboración propia

Para el siguiente paso que es probar el modelo, con el conjunto de prueba que se ha descrito anteriormente. Se realiza la evaluación del modelo mediante la función que se observa en la ***Figura 13***.

Figura 13
Evaluación del modelo

```
[43] # Evaluate the loaded model on the test data
     test_loss, test_acc = model.evaluate(
         test_generator,   # Data generator providing batches of test images and labels
         steps=len(test_generator)  # Number of steps (batches) to draw from the generator for evaluation
     )

     # Print the accuracy of the model on the test dataset
     print(f'Test Accuracy: {test_acc}')

     # Print the loss of the model on the test dataset
     print(f'Test Loss: {test_loss}')

89/89 ━━━━━━━━━━━━━━━━━━━━ 36s 400ms/step - accuracy: 0.8590 - loss: 0.4134
Test Accuracy: 0.8588652610778809
Test Loss: 0.4464718699455261
```

Fuente: Elaboración propia

Con la finalidad de que el código que se ha descrito anteriormente pueda ser fácilmente replicado, se ha creado el ***anexo C*** en él se adjunta el código con el que se ha entrenado y probado el modelo. Igualmente se ha guardado en el repositorio de *Google drive*[18]

En la siguiente tabla a modo de resumen se presenta la arquitectura del modelo, en esta se observan las capas del modelo, las formas de salida y el número de parámetros en cada capa. La capa de entrada (*Inputlayer*) define la entrada al modelo, que en este caso caso corresponde a una resolución de 224x224 y 3 canales de color (RGB). Se observan los modelos pre-entrenados descritos anteriormente (i) *MobileNetV2*, una red para la clasificación de imágenes pre-entrenada, su salida es de tamaño (None, 7, 7, 1280), lo que significa que después de procesar la imagen, reduce su dimensionalidad a un mapa de

18
https://drive.google.com/drive/folders/1PxJWT1_AQiwn6BoGkEkYzkc_9z5RHzwR?usp=drive_link

características de 7x7 con 1280 canales (ii) *densenet169* que es otro modelo preentrenado, que también es una arquitectura de red profunda pero más densa. Su salida es de tamaño (*none*, 7, 7, 1664). La capa concatenación, concatena la salida de los dos modelos pre entrenados, lo que produce un vector de tamaño (*none, 2944*) (iii) la capa final (*dense_2*), tiene 2 unidades de activación con *softmax,* produce una salida con una probabilidad para las dos clases (positivo y negativo).

Tabla 3

Arquitectura del modelo

```
model.summary()
```

Model: "functional"

Layer (type)	Output Shape	Param #	Connected to
input_layer (InputLayer)	(None, 224, 224, 3)	0	-
mobilenetv2_1.00_224 (Functional)	(None, 7, 7, 1280)	2,257,984	input_layer[0][0]
densenet169 (Functional)	(None, 7, 7, 1664)	12,642,880	input_layer[0][0]
global_average_pooling2d (GlobalAveragePooling2D)	(None, 1280)	0	mobilenetv2_1.00_224[…
global_average_pooling2d… (GlobalAveragePooling2D)	(None, 1664)	0	densenet169[0][0]
flatten (Flatten)	(None, 1280)	0	global_average_poolin…
flatten_1 (Flatten)	(None, 1664)	0	global_average_poolin…
concatenate (Concatenate)	(None, 2944)	0	flatten[0][0], flatten_1[0][0]
batch_normalization (BatchNormalization)	(None, 2944)	11,776	concatenate[0][0]
dense (Dense)	(None, 256)	753,920	batch_normalization[0…
dropout (Dropout)	(None, 256)	0	dense[0][0]
batch_normalization_1 (BatchNormalization)	(None, 256)	1,024	dropout[0][0]
dense_1 (Dense)	(None, 128)	32,896	batch_normalization_1…
dropout_1 (Dropout)	(None, 128)	0	dense_1[0][0]
dense_2 (Dense)	(None, 2)	258	dropout_1[0][0]

Total params: 15,700,740 (59.89 MB)
Trainable params: 793,474 (3.03 MB)
Non-trainable params: 14,907,264 (56.87 MB)
Optimizer params: 2 (12.00 B)

Fuente: Elaboración propia

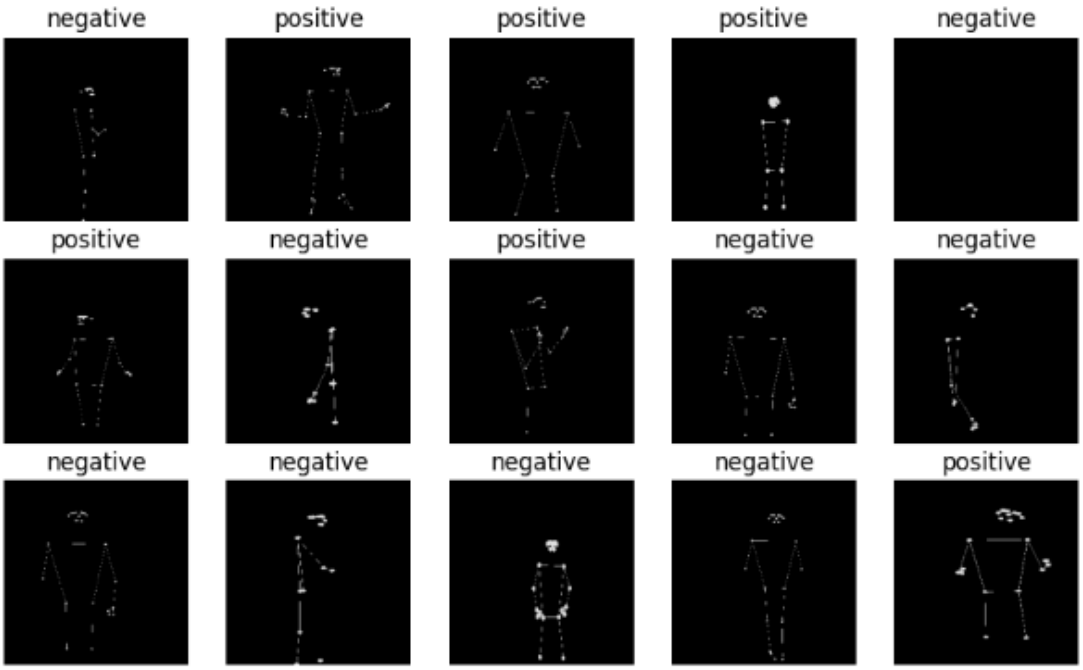
A continuación, en la ***Figura 14*** se observa una muestra del lote (*batch)*[19] que se ha usado en este proyecto. Se observa un ejemplo del lote del total de datos que es utilizado para entrenar el modelo en una única iteración. Este proceso se hace para inspeccionar

[19] https://pytorch.org/docs/stable/data.html

visualmente un lote de imágenes durante el entrenamiento del modelo, y comprobar que las etiquetas y el conjunto de imágenes es correcto.

Figura 14

Ejemplo del lote usado en el proyecto



Fuente: Elaboración propia

4.3.4. Evaluación del modelo

El proceso de evaluación del modelo tiene como objetivo medir el rendimiento y su capacidad para generalizar a nuevos datos. Luego de entrenar el modelo y su posterior evaluación con el conjunto de prueba, se han escogido las siguientes métricas de evaluación *accuracy*[20], *precision*[21], y *recall*[22].

En un sistema de clasificación binario como es el caso de este proyecto, la métrica *accuracy* se utiliza para medir la proporción que existe entre las predicciones correctas realizadas por

[20] https://keras.io/api/metrics/accuracy_metrics/#accuracy-class
[21] https://keras.io/api/metrics/classification_metrics/#precision-class
[22] https://keras.io/api/metrics/classification_metrics/#recall-class

el modelo en comparación con el total de predicciones realizadas. En la *Ecuación 1* se observa la definición de esta métrica.

Ecuación 1 $$\boldsymbol{accuracy} = \frac{\boldsymbol{número\ de\ predicciones\ correctas}}{\boldsymbol{número\ total\ de\ predicciones}}$$

Lo que es igual a

Ecuación 2 $$\boldsymbol{accuracy} = \frac{\boldsymbol{TP+TN}}{\boldsymbol{TP+TN+FP+FN}}$$

Donde:

TP (*True Positives*): Predicciones correctas de la clase positiva.

TN (*True Negatives*): Predicciones correctas de la clase negativa.

FP (*False Positives*): Predicciones incorrectas donde el modelo predijo la clase positiva incorrectamente.

FN (*False Negatives*): Predicciones incorrectas donde el modelo no predijo la clase positiva cuando debería haberlo hecho.

Para el presente proyecto, el modelo ha arrojado un *accuracy de* ***0.86,*** lo que significa que el modelo tiene una precisión del 86% al hacer predicciones. En este proyecto se busca que el sistema de evaluación de las posturas sea lo más preciso posible. Un 86% de precisión se puede considerar como un buen desempeño para el sistema. En la literatura se han reportado otros sistemas que pretendían evaluar las posturas de los presentadores como por ejemplo en (Ochoa et at., 2018) con el sistema RAP (Retroalimentación Automática de Presentaciones), reportó un *accuracy* del 80%. Siendo este sistema uno de los referentes en la literatura sobre evaluación de presentaciones automáticas de presentaciones orales, se puede inferir que el modelo obtenido para la evaluación de posturas en el presente proyecto tiene un resultado aceptable.

La métrica *precision* se define por la proporción de predicciones correctas entre todas las predicciones positivas que hizo el modelo. Dicho de otra forma, responde a la siguiente pregunta: De todas las posturas que ha predicho el modelo como positivas, ¿cuántas lo son realmente?

Se define por la siguiente formula:

Ecuación 3 $$\boldsymbol{Precision = \frac{TP}{TP+FP}}$$

Donde:

TP *(True Positives):* Número de posturas correctamente predichas como positivas.

FP *(False Positives):* Número de posturas incorrectamente predichas como positivas (es decir, se predijeron como positivas, pero en realidad eran negativas).

Para el presente proyecto, el modelo ha arrojado una *precision* de ***0.85***. Este valor sigue siendo aceptable dados los resultados encontrados en la literatura (Ochoa et at., 2018).

La métrica *recall,* también llamada sensibilidad, se utiliza para medir la capacidad del modelo de identificar correctamente las posturas positivas entre todas las posturas s que realmente lo son. Es decir, el *recall* contesta el siguiente interrogante: De todas las posturas que realmente pertenecen a la clase positiva, ¿cuántas fueron correctamente identificadas por el modelo?

La *Ecuación 4* muestra la formula asociada.

Ecuación 4 $$\boldsymbol{Recall = \frac{TP}{TP+FN}}$$

Donde:

TP (*True Positives*): Número de posturas correctamente predichas como positivas.

FN (*False Negatives*): Número de posturas que son realmente positivas, pero fueron incorrectamente predichas como negativas por el modelo.

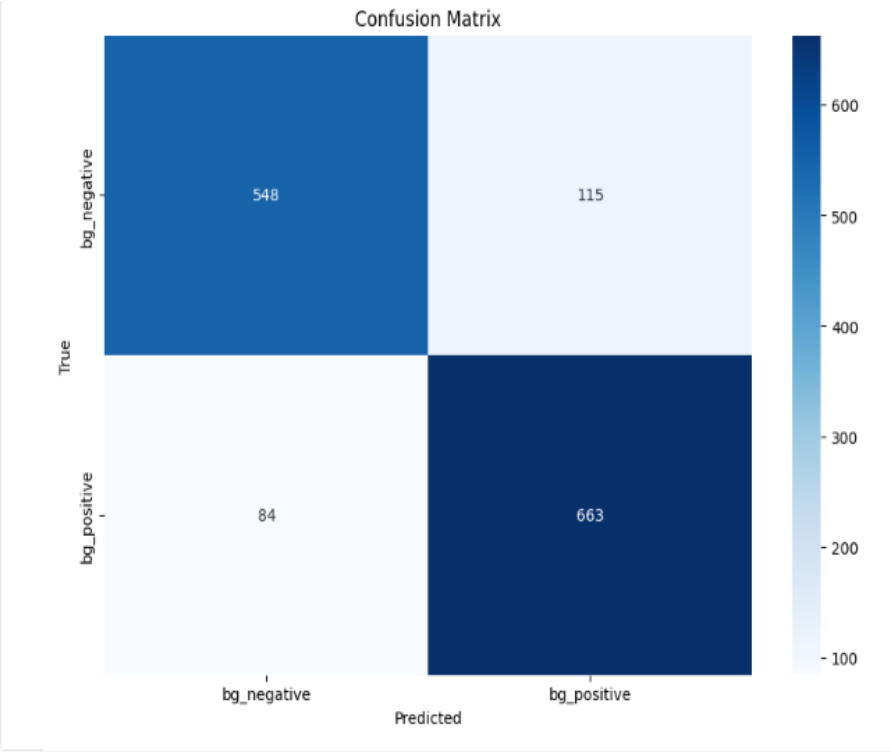
Para el presente proyecto, el modelo ha arrojado un *recall* de ***0.89.*** En la literatura se han reportado otros sistemas que pretendían evaluar las posturas de los presentadores como por ejemplo en (Ochoa et at., 2018) con el sistema RAP (Retroalimentación Automática de Presentaciones), reportó un *recall* del 93%, mientras que en (Alshammari, Abd Rahman, Arshad, & Albahri, 2023) con su sistema *Real-Time Robotic Presentation Skill,* desarrollaron un sistema de puntuación de habilidades de presentación en tiempo real, emplearon un análisis multimodal y el proceso de jerarquía analítica difusa *Delphi*. Este modelo específicamente para evaluar la detección del movimiento de las manos, arrojó una medida

de *precision* de 53% y *recall* del 95%. Dada la comparación de estos sistemas con el sistema presentado en el proyecto, las métricas obtenidas se encuentran en un rango aceptable.

Otra herramienta que se utilizó para evaluar el modelo fue la matriz de confusión, la cual muestra el rendimiento del modelo de clasificación, mostrando cómo se distribuyen las predicciones del modelo en relación con las etiquetas verdaderas de las clases. En la ***Figura 15*** se observa la matriz de confusión del modelo.

Figura 15

Matriz de confusión del modelo



Fuente: Elaboración propia

La matriz de confusión nos indica que el modelo en el conjunto de test que consta de 1410 imágenes, ha predicho correctamente con la etiqueta negativa 548 imágenes y ha predicho correctamente con la etiquete positiva 663 imágenes, lo que corresponde a los porcentajes obtenidos en las métricas presentadas anteriormente, y como se ha mencionado en el apartado

de las métricas, estos valores son aceptables para el presente proyecto, ya que contrastado con la literatura, los valores obtenidos en el modelo son muy similares.

Los sistemas de presentaciones orales tienen como objetivo encontrar un balance entre las métricas *precision* y *recall* ya que un valor alto en cualquiera de estas métricas, en algunos casos puede implicar un bajo rendimiento en la otra. En la ***Figura 16*** se muestra la gráfica de *accuracy* del modelo durante el entrenamiento y la prueba en función de los *epochs.*

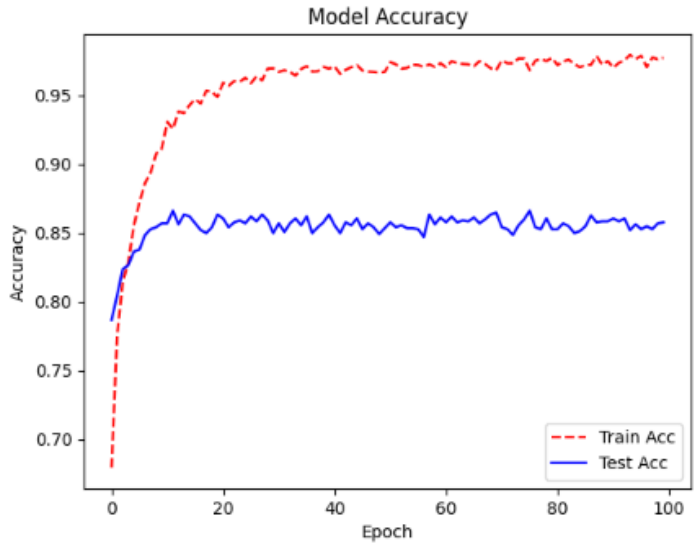
La curva roja muestra el rendimiento del modelo sobre el conjunto de entrenamiento, se observa que la precisión del entrenamiento aumenta rápidamente en la primeras *epochs,* casi alcanzando el 100% lo que significa que el modelo se está ajustando muy bien a los datos de entrenamiento.

La curva azul representa el rendimiento del modelo en el conjunto de prueba, se observa que se estabiliza en torno a una precisión del 85% y muestra algunas oscilaciones en torno a este valor. El hecho de que no suba significativamente después de las primeras *epochs* sugiere que el modelo ha aprendido los patrones generales del conjunto de datos, pero no mejora más allá de cierto punto en los datos de prueba.

Por lo que esta diferencia de precisiones sugiere que el sistema está sobreajustado *(overfitting)*, no obstante, el sistema muestra un buen rendimiento, aunque sugiere que se puede mejorar los signos de sobreajuste, tal vez realizando cambios en la arquitectura del modelo, generando un conjunto de datos mucho más grande con una muestra aleatoria, entre otros.

Figura 16

Gráfica de accuracy durante el entrenamiento.



Fuente: elaboración propia

4.3.5. Implementación de la interfaz final de usuario del sistema

Con la finalidad de cumplir con el objetivo de presentar un sistema que sea de fácil uso para la evaluación de posturas, se llevó a cabo la implementación de una interfaz muy sencilla. Como la idea final del sistema es poder evaluar una presentación, es decir un video y no solo una imagen, se le pidió de nuevo a los 26 participantes de este proyecto que realizaran un pequeño video de 3 minutos, en donde se les pedía que expusieran un tema de su elección y que lo hicieran de forma natural, se les explicó que el contenido verbal no sería evaluado, ya que para los objetivos de este proyecto no es relevante.

Para la implementación de esta parte del proyecto se desarrolló una interfaz utilizando *Tkinter*[23] es la librería GUI (*Graphical User Interface*) estándar en *Python*. Esta librería se ha escogido para este proyecto, debido a su fácil integración con Python, no es necesario instalar dependencias adicionales, su tasa de aprendizaje es muy buena, ya que tiene bastante documentación y una comunidad activa de usuarios lo que es muy útil para encontrar

[23] https://docs.python.org/3/library/tkinter.html

soluciones, es compatible con múltiples sistemas operativos como lo son Windows, macOS y Linux. En este proyecto, *Tkinter* se utiliza para construir la GUI que permite a los usuarios cargar imágenes o vídeos, ejecutar el modelo de estimación de pose y visualizar los resultados directamente dentro de la aplicación.

También se utilizó la biblioteca *Matplotlib*[24]*, cuya* función es crear visualizaciones estáticas, animadas e interactivas en *Python*. En este proyecto, *Matplotlib* se utiliza para generar gráficos de barras que visualizan las puntuaciones de pose positivas y negativas predichas por el modelo. Estos gráficos se muestran en la interfaz gráfica *Tkinter*. Junto con *Matplotlib, se empleó* la librería de visualización *Seaborn*[25] de *Python*, que proporciona una interfaz de alto nivel para dibujar gráficos estadísticos atractivos e informativos. Aunque su uso en este proyecto es limitado, *Seaborn* se incluye para mejorar el atractivo visual de los gráficos generados.

Para facilitar la lectura de la retroalimentación automática, que se presenta al orador al finalizar el análisis de su video *(*tal y como se hizo por (Ochoa et at., 2018*))*, se utilizó la biblioteca de *Python Pandas*[26]*, que* es una herramienta de código abierto para el análisis y la manipulación de datos basada en *Python*. Se utiliza en este proyecto para gestionar y analizar los datos de retroalimentación almacenados en archivos Excel, que luego se utilizan para proporcionar mensajes de retroalimentación basados en las predicciones del modelo.

De igual forma la biblioteca *Pillow*[27]*,* ampliamente utilizada para la manipulación y procesamiento de imágenes, permite cargar imágenes desde archivos en formatos populares como JPEG, PNG, BMP, GIF y muchos otros. En este proyecto, *Pillow* se utiliza para convertir imágenes entre diferentes formatos y prepararlas para su visualización en la interfaz gráfica *Tkinter*.

A continuación, en ***Figura 17*** se presenta el resultado final de la interfaz gráfica, su código fuente se encuentra en el mismo repositorio del *Google drive*[28].

[24] https://matplotlib.org/
[25] https://seaborn.pydata.org/
[26] https://interactivechaos.com/es/manual/tutorial-de-pandas/tutorial-de-pandas
[27] https://pillow.readthedocs.io/en/stable/
[28] https://bit.ly/3XLnmh7

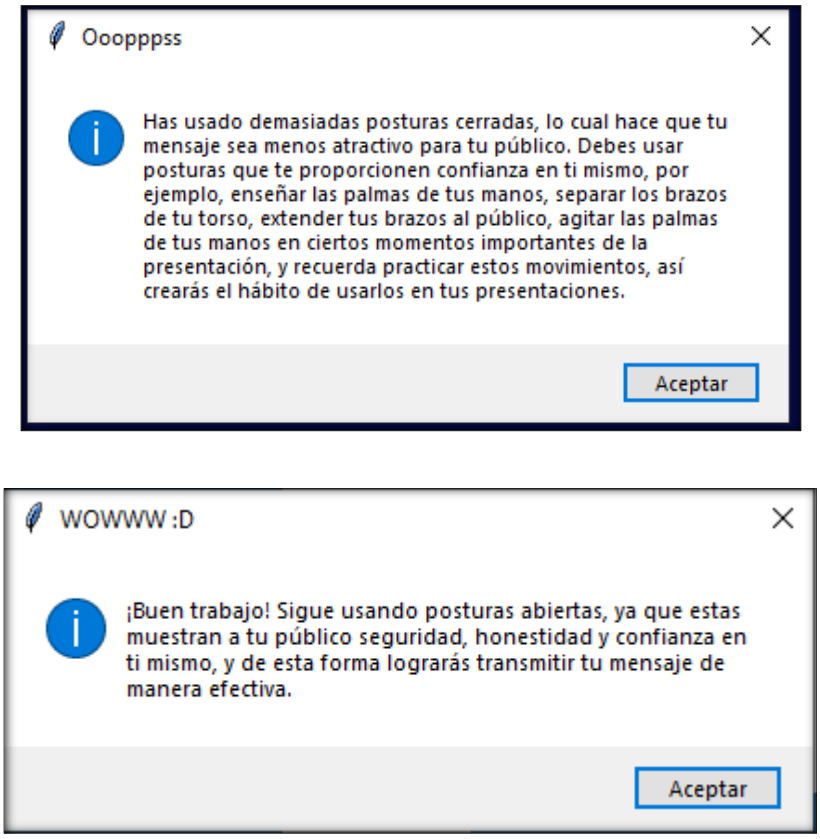
Figura 17

Interfaz gráfica del sistema

En la ***Figura 17*** se muestra un ejemplo de evaluación del sistema, se ha cargado una imagen de un participante en la postura "brazos detrás del cuerpo" y se ha utilizado para evaluar el sistema, el cual ha arrojado que efectivamente es una pose negativa o cerrada. Automáticamente el sistema arroja el mensaje que se presenta en la ***Figura 18***, este mensaje se ha cargado de forma predeterminada en el sistema, mediante un archivo de *Excel,* y el sistema muestra el mensaje dependiendo si el resultado de la postura es positiva o negativa.

Figura 18

Retroalimentación dada al estudiante cuando se detecta que ha usado una postura negativa o cerrada, o cuando el estudiante ha realizado una postura abierta y positiva

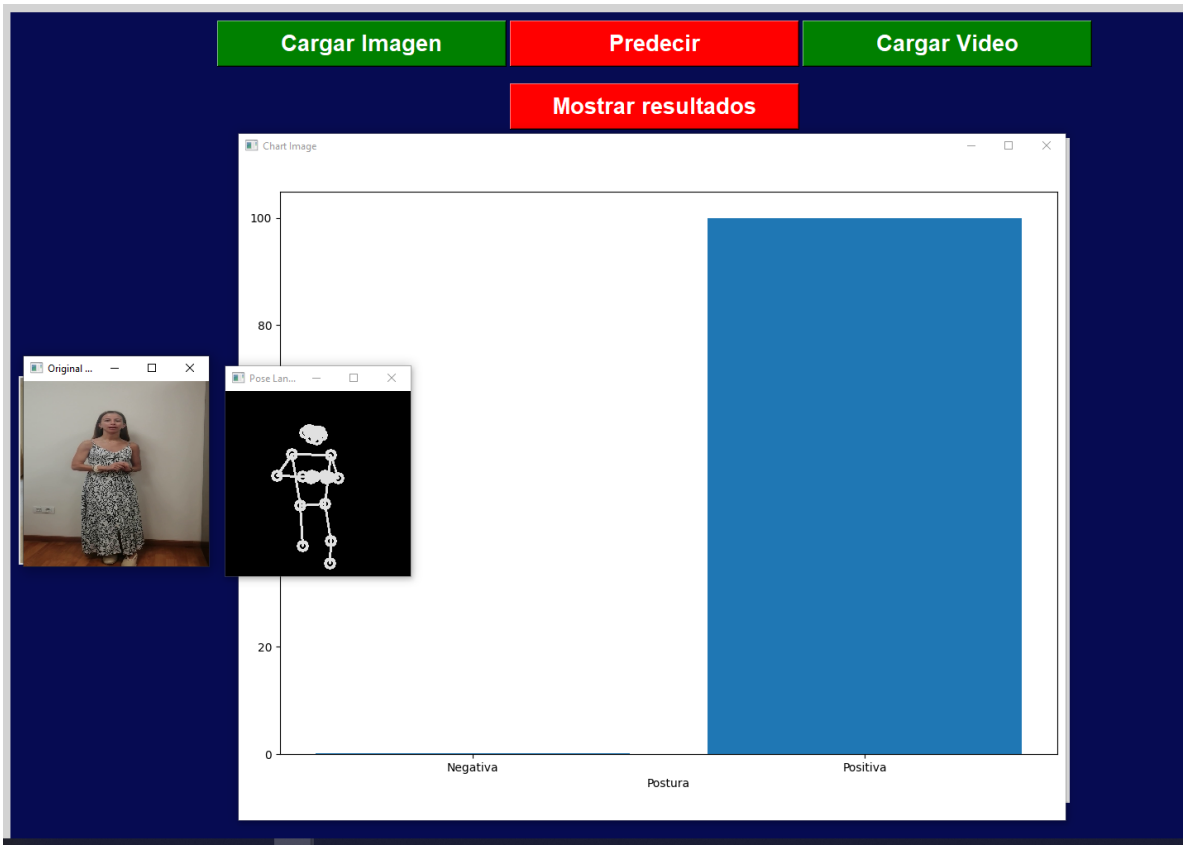


Para la evaluación de los videos, el proceso se hace por cada fotograma, todos los resultados se guardan en dos variables de contador diferentes. Variable1 (*positive_score*) es para guardar todas las predicciones positivas. Variable2 *(negative_score*) es para guardar todas las predicciones negativas. Después de procesar todas las variables, se verifica si el conteo total de *positive_score* es mayor que el *negative_score*. Con esta función, luego se agrega la retroalimentación. Por ejemplo, si 70 de cada 100 fotogramas son positivos, entonces el puntaje positivo es 70%, o viceversa.

En la ***Figura 19*** se observa un ejemplo de cómo el sistema predice las posturas del video, es capaz de procesar cada fotograma y predecir la pose, y al finalizar la predicción de las posturas de todo el video, almacena todas las predicciones tanto positivas como negativas, luego comprueba la totalidad de las predicciones, y el resultado arrojado es el que contenga

el mayor número de predicciones, ya sea mayor número de predicciones de posturas positivas o viceversa.

Figura 19
Interfaz utilizada para la evaluación de las posturas *del presentador*



5. Análisis de los resultados

Este capítulo expone los hallazgos significativos obtenidos tras la implementación de los algoritmos de aprendizaje automático. Estos algoritmos han sido diseñados para interpretar y analizar con precisión los patrones del lenguaje corporal durante presentaciones orales. La investigación se enfoca en cómo estas técnicas pueden contribuir a mejorar la comunicación no verbal, elemento clave en la transmisión efectiva de ideas y emociones.

El desarrollo de este proyecto ha llevado a la creación de un sistema para la evaluación automática de presentaciones orales mediante el análisis de posturas corporales. A continuación, se abordan los hitos clave del proyecto, incluyendo la realización del modelo de predicción de posturas, la construcción de la interfaz para el reconocimiento de posturas y la evaluación del mismo.

Para dar solución al objetivo principal de este proyecto se ha llevado a la creación de un sistema potencialmente: útil y con una interfaz eventualmente amigable al usuario, para que el estudiante, profesor o investigador, pueda hacer uso de esta herramienta. El sistema ha incluido la creación de un modelo de *machine learning,* mediante el cual se lograse realizar la tarea de la evaluación automática de posturas. En el capítulo 4 se ha descrito todo el proceso que se ha llevado a cabo para cumplir con el objetivo general del presente proyecto, que consistía en desarrollar un sistema de evaluación del lenguaje corporal utilizando técnicas de *machine learning.* Al finalizar este capítulo se han discutido los resultados de la evaluación del modelo y se ha contrastado el resultado con algunos modelos existentes en la literatura, lo que ha arrojado que este el modelo de clasificación tiene un buen rendimiento en el conjunto de prueba, obteniendo una precisión de alrededor del 85% lo cual es un dato aceptable según lo visto en la literatura y para efectos experimentales de este proyecto, cumple con la premisa de tener un desempeño aceptable. Con respecto a las otras métricas obtenidas como *recall* del 89%, también se puede observar que es un rendimiento aceptable dentro de lo que se espera que responda el modelo

Uno de los mayores desafíos e inconvenientes que se ha tenido a lo largo del desarrollo del proyecto ha sido la obtención de la muestra (*dataset*), al requerir una gran cantidad de muestras, este proceso requería una alta demanda de tiempo y participantes, lo que ha sido limitado en el desarrollo del sistema. Aun así, se exploraron las alternativas que ofrece la tecnología y se utilizó la técnica de aumento de datos, y se pudo obtener un mejor desempeño del sistema, hay que tener en cuenta que esto sigue siendo un problema, ya que estas imágenes generadas son artificiales y para poder tener una mejor generalización del problema ideal sería tener una muestra que incluya estudiantes de diferentes regiones del mundo.

Antes de crear un *dataset* propio, se investigó qué conjuntos de datos ya creados y etiquetados eran de dominio público y que se adaptaran al contexto de las presentaciones orales. El estudio arrojó únicamente un conjunto de datos disponible en la web, se trata de *Multi-Sensor Presentation (NUSMSP) dataset* descrito en (Gan, T., et al., 2019). Este conjunto de datos fue creado específicamente para analizar presentaciones utilizando una combinación de sensores como acelerómetros y giroscopios para registrar el movimiento y la orientación. Consta de 51 presentaciones únicas, realizadas por estudiantes de pregrado, posgrado y algunos investigadores. Las anotaciones de este conjunto de datos se hicieron manualmente, en clips de 10 segundos, utilizando una rúbrica de evaluación diseñada por ellos mismos que incluye el análisis del audio, el lenguaje corporal, la atención del presentador y el compromiso con la audiencia. Las principales desventajas y las razones por las cuales no se pudo utilizar este *dataset* son: (i) el conjunto de datos era pequeño, solo constaba de 51 presentaciones, no era una cantidad apropiada para entrenar modelos de *deep learning* (ii) tanto los sensores utilizados como el entorno en el que fue desarrollado, no eran fácilmente replicables ni adaptables, (iii) el etiquetado y los criterios del conjunto de datos no correspondían con el modelo planteado en este proyecto, (iv) el proceso de extracción de los puntos proporcionados por el sensor *Kinect* y la posterior anotación de las etiquetas para reconstruir tanto las posturas positivas como las negativas, así como la identificación de las necesarias para este proyecto, resultó ser extremadamente complejo debido al alto nivel de precisión y detalle exigido.

Los resultados indican que el algoritmo tiene potencial para identificar de una manera eficaz las posturas deseadas. Las métricas alcanzadas en esta fase inicial del proyecto pueden ser

mejorables construyendo un conjunto de datos más grande, de esta forma se podrá llevar a cabo con mayor precisión las predicciones de las posturas.

La clasificación efectiva de posturas y gestos es fundamental en el campo del aprendizaje automático. En este caso, se utilizaron redes neuronales de convolución (CNN), que son un tipo de red neuronal artificial especializada en procesar datos con una estructura en forma de cuadrícula, como imágenes. Las CNN son efectivos para este propósito porque pueden identificar y extraer características importantes de los datos de entrada, que en este contexto son los *landmarks* o puntos de referencia capturados por *MediaPipe*, una solución de visión por computadora. Al analizar estos puntos de referencia, las CNN pueden clasificar con precisión las posturas y gestos en categorías específicas, lo que es crucial para aplicaciones como el reconocimiento de gestos en tiempo real o la interacción hombre-máquina. Esta metodología sirvió para llevar a cabo el proceso de la clasificación de las posturas, lo que demuestra la eficacia de las CNN en tareas de clasificación complejas.

La implementación de este algoritmo de evaluación de presentaciones ofrece una solución eficiente y objetiva para medir aspectos no verbales de la comunicación, lo cual es crucial en un entorno académico y profesional. La capacidad de cuantificar la presencia de posturas específicas pudiere llegar a proporcionar una retroalimentación valiosa para los presentadores, permitiéndoles mejorar su lenguaje corporal y, por ende, su efectividad en la comunicación oral.

Los resultados de esta evaluación están apoyados en la literatura sobre la importancia de la comunicación no verbal en las presentaciones orales. Las posturas positivas, como una postura erguida y gestos abiertos, pueden transmitir confianza, seguridad y entusiasmo. Por otro lado, las posturas negativas, como encorvarse, tener posturas cerradas, pueden indicar inseguridad, falta de preparación y desinterés.

Es importante destacar que las posturas no son el único factor que determina la calidad de una presentación oral. Otros factores como el contenido de la presentación, la organización, la claridad del discurso y la fluidez verbal también son importantes. Sin embargo, las posturas pueden jugar un papel importante en la primera impresión que se genera en la audiencia y en la forma en que se recibe el mensaje.

El proyecto alcanzó importantes hitos, pero no estuvo exento de obstáculos, en las primeras pruebas del modelo se observó que la variabilidad en la iluminación y en los fondos de las imágenes podía comprometer la bondad del algoritmo encargado de identificar puntos de referencia. Para superar estos retos, se transformó el espacio de análisis de los puntos de referencia, llevándolos a una imagen en negro, la cual evitaría que objetos anejos a la extracción de la pose pudiese causar cualquier error.

Otra de las limitaciones de este trabajo ha sido el tamaño reducido del conjunto de datos creado. La poca variabilidad y el tamaño limitado de la muestra impiden obtener un modelo suficientemente generalizado para su uso a nivel académico. Aún es necesario contar con un conjunto de datos mucho más grande. Además, se observa un desequilibrio de clases, ya que el conjunto de datos positivo está sobrerrepresentado, mientras que el conjunto de datos negativo es escaso. Esto podría generar sesgo, ya que el modelo tiende a aprender más sobre la clase dominante.

6. Conclusiones

El objetivo central de este proyecto fue la creación de un sistema capaz de identificar y valorar el lenguaje corporal en presentaciones orales, utilizando para ello métodos de aprendizaje automático. Durante el proceso investigativo, se establecieron y cumplieron metas particulares que, en conjunto, facilitaron la consecución del propósito principal. Las conclusiones alcanzadas, basadas en los resultados obtenidos de cada meta específica, son fundamentales para comprender el impacto y la eficacia del sistema desarrollado. Estas conclusiones no solo reflejan el cumplimiento de los objetivos individuales, sino que también demuestran la integración efectiva de las técnicas de *machine learning* en la evaluación del lenguaje no verbal, un componente esencial en la comunicación humana.

Identificación y Recopilación de Datos de Entrenamiento: El primer paso fundamental y primer objetivo en este proyecto fue la identificación y recopilación del conjunto de datos. Estos datos consistieron en imágenes de oradores ejecutando posturas y gestos específicos. Luego, se implementó un algoritmo utilizando *MediaPipe* para extraer *landmarks* y llevarlos a una imagen con fondo negro la cual proporcionó la información esencial para el entrenamiento de modelos de *machine learning*.

Principios de Visión por Computadora y Técnicas de Detección de Puntos de Articulación: El objetivo de comprender los principios de visión por computadora y el uso de técnicas como *OpenPose* se logró mediante la implementación de *MediaPipe*. Esta biblioteca demostró ser una herramienta poderosa para la detección y seguimiento de puntos de articulación del cuerpo en videos, proporcionando *landmarks* cruciales para la evaluación del lenguaje corporal en presentaciones orales.

Definición y Aplicación de Criterios de Clasificación: La definición y aplicación de criterios para clasificar gestos y posturas como "positivas" o "negativas" en el contexto de una presentación oral fue un aspecto crucial del proyecto. Se estableció un sistema de evaluación mediante expertos y con la revisión sistemática de la literatura que sirvió como guía para la

clasificación de las posturas detectadas. Este paso proporcionó la base para la creación de modelos de *machine learning* capaces de realizar clasificaciones automáticas.

Aplicación de Técnicas de Machine Learning: La aplicación de técnicas de *machine learning*, específicamente el uso de redes neuronales de convolución (CNN), permitió entrenar modelos capaces de clasificar posturas. Estos modelos demostraron ser efectivos para generalizar a nuevas instancias y adaptarse a condiciones variadas.

Evaluación Crítica de la Metodología y Resultados: Se llevó a cabo una evaluación crítica de la metodología, los resultados y las conclusiones del proyecto. Se reconocieron los desafíos, como la variabilidad en las condiciones de iluminación, y se identificaron posibles mejoras, como la expansión del conjunto de datos y la inclusión de más categorías de posturas. La comunicación efectiva de estos hallazgos es crucial para fomentar el desarrollo continuo y la aplicación futura del sistema.

Con la finalidad de evaluar sí la interfaz cumple con un diseño de uso fácil, se le ha preguntado a los 26 participantes mediante una encuesta se incluye en el ***anexo D***, que tan fácil es el uso de la interfaz, obteniendo los siguientes resultados: (i) los encuestados se encuentran en una edad entre 25 y 36 años repartidos equitativamente entre hombres y mujeres, (ii) el 32% de los encuestados manifestó tener experiencia básica con interfaces similares, mientras que un 31% manifestó que no había tenido ninguna experiencia el 21% experiencia moderada y el 16% avanzada, (iv) el 49% de los encuestados manifestó haber experimentado problemas con la funcionalidad de ciertos elementos de la interfaz, mientras que el 51% no encontró problemas, (v) con respecto a la facilidad de interacción con los botones, el 73% indicó fácil, el 20% neutral y el 7% difícil, (vi) en relación con problemas con la funcionalidad de la interfaz el 49% respondió si y el 51% no, (vii) en términos de satisfacción general el 68% se encuentra muy satisfecho y el 32% se mostró insatisfecho.

En conclusión, este proyecto ha cumplido en gran medida con los objetivos planteados, logrando la creación de un sistema que emplea visión por computadora y técnicas de machine learning para evaluar el lenguaje corporal en presentaciones orales. El sistema ofrece una herramienta valiosa para mejorar las habilidades de presentación y establece una base sólida para futuras investigaciones y aplicaciones en los campos de la inteligencia artificial y la

comunicación humana. No obstante, no se puede afirmar sobre su generalización, principalmente debido a la cantidad limitada de datos reales utilizados, lo que podría afectar la precisión del modelo al aplicarse en entornos más diversos. Para futuras investigaciones, sería fundamental ampliar el conjunto de datos, incluir una mayor variedad de contextos y presentadores, y mejorar el equilibrio de las clases, con el fin de desarrollar un sistema más robusto y adaptable a diferentes situaciones. A pesar de estas limitaciones, el sistema es de bajo coste y escalable, lo que lo hace accesible para cualquier persona interesada en utilizarlo y perfeccionarlo.

7. Líneas futuras de investigación

Para conseguir mejores desempeños y lograr una integración real con el sistema educativo, es necesario seguir investigando y adaptando los nuevos sistemas de evaluación de presentaciones automáticas, a continuación, se plantean algunas investigaciones que se pueden desarrollar en este campo.

1. **Evaluación de la efectividad de diferentes modalidades de retroalimentación**: Es necesario realizar estudios empíricos que comparen los efectos de la retroalimentación en tiempo real versus la retroalimentación posterior a la presentación, con el objetivo de determinar cuál es la más efectiva para mejorar las habilidades orales específicas.

2. **Mejorar la usabilidad y accesibilidad de los sistemas:** En futuros trabajos deberían centrarse en mejorar la usabilidad y reducir la intrusividad de los sistemas, facilitando su uso tanto por parte de estudiantes como de instructores, lo que aumentaría su adopción en entornos educativos.

3. **Estudios prolongados en entornos educativos reales:** La mayoría de los estudios hasta ahora se han realizado en entornos controlados. Sería valioso realizar estudios prolongados en contextos educativos reales para evaluar el impacto a largo plazo de estos sistemas en el desarrollo de habilidades de presentación. Tal y como se menciona, en (Ochoa, 2022), en este documento el autor plantea como próximos pasos a seguir para la investigación de Sistemas Multimodales de Retroalimentación Automatizada de Presentaciones Orales (OPAF) debería conectarse con los sistemas de gestión del aprendizaje existentes, en los que se pueden asignar sesiones de prácticas y almacenar las evaluaciones. El OPAF en su conjunto debería plantearse no como un sistema independiente, sino como una parte de un diseño instruccional más completo.

4. **Desarrollo de marcos multimodales compartidos**: Para futuros trabajos se debería tener en cuenta desarrollar marcos o plataformas compartidas que permitan a los investigadores reutilizar soluciones existentes y facilitar la integración de nuevas funcionalidades. Esto podría mejorar la escalabilidad y la adopción de los sistemas en ambientes educativos reales, como se plantea en (Ochoa, 2022), en lugar de compartir el código en bruto, los investigadores podrían compartir los sistemas predefinidos y los complementos dentro de estos marcos, de este modo se establecerían las mejores prácticas y se incrementaría la eficacia y eficiencia de los nuevos OPAF.

Referencias

Abadi, M., et al. (2015). *TensorFlow: Large-scale machine learning on heterogeneous systems*. [Software]. Disponible en https://www.tensorflow.org.

Alshammari, R. F. N.; Abd Rahman, A. H.; Arshad, H., & Albahri, O. S. (2023). Real-time robotic presentation skill scoring using multi-model analysis and fuzzy Delphi–analytic hierarchy process. *Sensors, 23*(24), 9619. https://doi.org/10.3390/s23249619

Baltrusaitis, T.; Ahuja, C., & Morency, L.-P. (2018). Multimodal machine learning: A survey and taxonomy. IEEE Transactions on Pattern Analysis and Machine Intelligence, 41(2), 423-443. https://doi.org/10.1109/TPAMI.2018.2798607

Baron, R. (1981). Mechanisms of human facial recognition. *International Journal of Man-Machine Studies*, 15(2), 137-178.

Barua, A.; Ahmed, M. U. & Begum, S. (2023). A Systematic literature review on multimodal machine learning: Applications, challenges, gaps and future directions, *IEEE Access*, 11, 14804-14831, doi: 10.1109/ACCESS.2023.3243854.

Behoora, I. & Tucker, C. (2015). Machine learning classification of design team members' body language patterns for real time emotional state detection. *Design Studies*, 39, 100–127. doi: 10.1016/j.destud.2015.04.003

Bhatt, P.; Sethi, A.; Tasgaonkar, V., et al. (2023). Aprendizaje automático para el análisis cognitivo conductual: conjuntos de datos, métodos, paradigmas y direcciones de investigación. *Brain Informatics*. 10, 18 (2023). https://doi.org/10.1186/s40708-023-00196-6

Cao, Z.; Hidalgo, G.; Simon, T.; Wei, S. & Sheikh, Y. (2021). OpenPose: Realtime multi-person 2D pose Estimation using part affinity fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(1). 172-186. doi: 10.1109/TPAMI.2019.2929257.

Caridakis, G.; Raouzaiou, A.; Bevacqua, E.; Mancini, M.,; Karpouzis, K.; Malatesta, L., & Pelachaud, C. (2007). Virtual agent multimodal mimicry of humans. *Language Resources and Evaluation*, 41(3-4), 367-388. https://doi.org/10.1007/s10579-007-9057-1

Carney, D.; Cuddy, A. & Yap, A. (2010). Power posing: Brief nonverbal Ddisplays affect neuroendocrine levels and risk tolerance. *Psychological Science*, 21(10), 1363–1368. doi:10.1177/0956797610383437.

Carter, B. & Luke, S. (2020). Best practices in eye tracking research. *International Journal of Psychophysiology*, 155, 49-62.

Castro, S. (2023). *Lenguaje no verbal y lenguaje corporal: Ejemplos y Técnicas*. Instituto Europeo de Psicología Positiva. Disponible en: https://www.iepp.es/lenguaje-no-verbal-corporal/

Chen, M. & Risen, J. (2010). How choice affects and reflects preferences: Revisiting the free-choice paradigm. *Journal of Personality and Social Psychology*. 99 (4): pp. 573 - 594.

Chen, L.; Feng, G.; Joe, J.; Leong, C. W.; Kitchen, C. & Lee, C. M. (2014). Towards automated assessment of public speaking skills using multimodal Ccues. *Proceedings of the 16th International Conference on Multimodal Interaction - ICMI '14*. doi:10.1145/2663204.2663265.

Chollet, F., et al. (2015). *Keras*. [Software]. Disponible en: https://keras.io

CEDEC (2024). *Rúbrica para evaluar una exposición oral. Materiales del REA Cuando vivían en blanco y negro del Proyecto EDIA*. CEDEC – INTEF.

Cialdini, R. (2021). *Influence, The Psychology of Persuasion. New and Expanded*. Harper Business.

Cummings, M. L.,; Roff, H. M.; Cukier, K.,; Parakilas, J., & Bryce, H. (2018). *Artificial intelligence and international affairs: Disruption anticipated (Chatham House Report)*. Royal Institute of International Affairs.

Del Río, M., & Martín-Gutiérrez, J. (2020). Evaluation of full-body gestures performed by individuals with Down syndrome: Proposal for designing user interfaces for all based on Kinect sensor. *Sensors*, 20(14), 3930. https://doi.org/10.3390/s20143930

Domínguez, F.; Ochoa, X.; Zambrano, D.; Camacho, K. & Castells, J. (2021). Scaling and adopting a multimodal learning analytics application in an Iinstitution-wide setting. *IEEE Transactions on Learning Technologies*, 14(3), 400-414. doi: 10.1109/TLT.2021.3100778.

D'Mello, S. & Graesser, A. (2010). Multimodal semi-automated affect detection from conversational cues, gross body language, and facial features. *User Modeling and User-Adapted Interaction*, 20(2), 147–187. doi:10.1007/s11257-010-9074-4.

Gan, T.; Li, J., Wong, Y., & Kankanhalli, M. S. (2019). A multi-sensor framework for personal presentation analytics. *ACM Transactions on Multimedia Computing, Communications, and Applications* (TOMM), 15(2), Article 30. https://doi.org/10.1145/330094

Goodfellow, I.; Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.

Gulati, A. P. (2022). Pose detection in image using Media Pipe Library. *Analytics Vidhya*.

Gunes, H., & Schuller, B. W. (2013). Categorical and dimensional affect analysis in continuous input: Current trends and future directions. *Image and Vision Computing*, 31(2), 120-136. https://doi.org/10.1016/j.imavis.2012.06.016

Hanani, A.; Mohammad, A.; Bazbus, W. & Salameh, S. (2017). Automatic Estimation of Presentation Skills Using Speech, Slides and gestures, *International Conference on Speech and Computer*. http://dx.doi.org/10.1007/978-3-319-66429-3_17

Harris, C., et al. (2020). Array programming with NumPy. *Nature*, 585(7825), 357–362. https://doi.org/10.1038/s41586-020-2649-2

Hidalgo, G. (2019). *OpenPose: Whole-Body Pose Estimation*. Carnegie Mellon University.

Holland, E.; Wolf, E. B.; Looser, C. & Cuddy, A. (2017). Visual attention to powerful postures: People avert their gaze from nonverbal dominance displays. *Journal of Experimental Social Psychology*, 68, 60–67. doi:10.1016/j.jesp.2016.05.001.

Hsiao, K. & Rashvand, H. (2011). Integrating body language movements in augmented reality learning environment. *Human-centric Computing and Information Sciences*, 1, 1-10.

Hunter, J. (2007). Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, 9(3), 90–95. https://doi.org/10.1109/MCSE.2007.55.

IBM (2020). *Descubra cómo las redes neuronales convolucionales utilizan datos tridimensionales para tareas de clasificación de imágenes y reconocimiento de objetos*. IBM. https://www.ibm.com/es-es/topics/convolutional-neural-networks

Itseez (2015). *Open Source Computer Vision Library*. Disponible en: https://github.com/itseez/opencv

Jaramillo, N.; Santacruz, J. & Bolaños, M. (2013). Metodología para la construcción de algoritmos, estructurada en el procesamiento de información - Teoría ACT de Anderson. Encuentro Internacional de educación en Ingeniería. https://doi.org/10.26507/ponencia.1508

Jegham, I.; Ben Khalifa, A.; Alouani, I. & Ali Mahjoub, M. (2020). Vision-based human action recognition: An overview and real-world challenges. *Forensic Science International: Digital Investigation,* 32, 200901, https://doi.org/10.1016/j.fsidi.2019.200901

Jia, J., He, Y., & Le, H. (2020). A multimodal human-computer interaction system and its application in smart learning environments. In Blended learning. Education in a smart

learning environment: 13th International Conference, ICBL 2020, Bangkok, Thailand, August 24–27, 2020, Proceedings (Vol. 12218, pp. 3–14). Springer. https://doi.org/10.1007/978-3-030-51968-1_1

Kaur, P.; Krishan, K.; Sharma, S. K. & Kanchan, T. (2020). Facial-recognition algorithms: A literature review. *Medicine, Science and the Law*, 60(2), 131-139.

Kim, J.-W., Choi, J.-Y., Ha, E.-J., & Choi, J.-H. (2023). Human pose estimation using MediaPipe Pose and optimization method based on a humanoid model. *Applied Sciences, 13*(4), 2700. https://doi.org/10.3390/app13042700

Knapp, M. L., Hall, J. A., & Horgan, T. G. (2013). *Nonverbal communication in human interaction* (8th ed.). Cengage Learning.

Kotsiantis, S. B.; Zaharakis, I. D., & Pintelas, P. E. (2007). Supervised machine learning: A review of classification techniques. In *Emerging artificial intelligence applications in computer engineering* (Vol. 160, pp. 3-24). IOS Press. https://datajobs.com/data-science-repo/Supervised-Learning- [SB-Kotsiantis].pdf

Laborda, X. (2019). *Claves de la comunicación oral. Prácticas para el orador afable*. Editorial UOC

Lubienetzki, U., & Schüler-Lubienetzki, H. (2022). *How we talk to each other: The messages we send with our words and body language*. Springer.

Lugaresi, C., et al. (2019). MediaPipe: A framework for building perception pipelines. *arXiv preprint* arXiv:1906.08172.

McNeill, D. (2005). *Gesture and thought.* University of Chicago Press.

Mehrabian, A. (1981). *Silent messages: implicit communication of emotions and attitudes*. 2da Ed. Wadsworth Publishing Company.

Mehrabian, A. (2017). *Nonverbal communication.* Transaction Publishers.

Meng, T.; Jing, X.; Yan, Z. & Pedrycz, W. (2020). A survey on machine learning for data fusion. *Information Fusion*, 57, 115-129, https://doi.org/10.1016/j.inffus.2019.12.001

Mittelstadt, B. D.; Allo, P.,; Taddeo, M.,; Wachter, S., & Floridi, L. (2016). The ethics of algorithms: Mapping the debate. *Big Data & Society*, 3(2), 2053951716679679. https://doi.org/10.1177/2053951716679679

Murata, T. & Shin, J. (2014). Hand gesture and character recognition based on kinect sensor. *International Journal of Distributed Sensor Networks,* 10(7). doi:10.1155/2014/278460

Noda, K.; Arie, H.; Suga, Y. y& Ogata, T. (2014). Multimodal integration learning of robot behavior using deep neural networks. *Robotics and Autonomous Systems, 62*(6), 721-736, https://doi.org/10.1016/j.robot.2014.03.003

Ochoa, X.; Domínguez, F.; Guamán, B.; Maya, R.; Falcones, G., & Castells, J. (2018). The RAP system: Automatic feedback of oral presentation skills using multimodal analysis and low-cost sensors. *In Proceedings of the 8th International Conference on Learning Analytics and Knowledge (LAK '18)* (pp. 360–364). Association for Computing Machinery. https://doi.org/10.1145/3170358.3170406

Ochoa, X. (2022). Multimodal Ssystems for qutomated oral Ppresentation Ffeedback: A comparative Aanalysis. En: Giannakos, M.; Spikol, D.; Di Mitri, D.; Sharma, K.; Ochoa, X.; Hammad, R. (Eds) *The Multimodal Learning Analytics Handbook*. Springer. doi: 10.1007/978-3-031-08076-0_3

Ojeda-Castelo, J. J.,; Capobianco-Uriarte, M. d. L. M.; Piedra-Fernandez, J. A., & Ayala, R. (2022). A survey on intelligent gesture recognition techniques. *IEEE Access, 10*, 87135-87156. https://doi.org/10.1109/ACCESS.2022.3199358

Pan, S. J., & Yang, Q. (2016). A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10), 1345-1359. https://doi.org/10.1109/TKDE.2009.191

Pantic, M.; Pentland, A.; Nijholt, A., & Huang, T. (2006). Human computing and machine understanding of human behavior: A survey. In *Proceedings of the 8th international conference on multimodal interfaces (ICMI '06)* (pp. 239–248). Association for Computing Machinery. https://doi.org/10.1145/1180995.1181044

Paszke, A., et al. (2019). PyTorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems,* 32, 8024-8035. Disponible en: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

Pease, A., & Pease, B. (2019). *The definitive book of body language: The hidden meaning behind people's gestures and expressions*. Bantam Books.

Pedregosa, F., et al. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830.

Petty, R.; & Cacioppo, J. (1986). The elaboration likelihood model of persuasion. *Advances in Experimental Social Psychology*, 123-205. doi:10.1016/s0065-2601(08)60214-2

Pons-Moll, G.; Baak, A.; Helten, T.; Müller, M.; Seidel, H. P.; & Rosenhahn, B. (2010). Multisensor fusion for 3D full-body human motion capture. *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 663-670.

Pham, H.; Khoudour, L.; Crouzil, A.; Zegers, P., & Velastín, S.A. (2022). Video-based Human Action Recognition using Deep Learning: A Review. *ArXiv, abs/2208.03775*.

Qiao, S.; Wang, Y.; & Li, J. (2017). Real-time human gesture grading based on OpenPose. 2017 *10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*. doi:10.1109/cisp-bmei.2017.8301910

Rayón, A.; Guenaga, M., & Núñez, A. (2014). Supporting competency-assessment through a learning analytics approach using enriched rubrics. In Proceedings of the *Second International Conference on Technological Ecosystems for Enhancing Multiculturality* (TEEM '14) (pp. 291-298). Association for Computing Machinery. https://doi.org/10.1145/2669711.2669913

Richardson, D. & Spivey, M. (2004). Eye tracking: Characteristics and methods. En Wnek, G. Bowlin, G. (Eds.) *Encyclopedia of biomaterials and biomedical engineering*, 1028-1042. Informa Healthcare USA.

Sánchez-Mena, A., & Martí-Parreño, J. (2017). Ethical issues in the use of learning analytics and big data for educational purposes. *International Education Studies*, 10(3), 52-61. https://doi.org/10.5539/ies.v10n3p52

Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1), 60. https://doi.org/10.1186/s40537-019-0197-0

Singh, V., & Misra, A. K. (2020). Plant leaf disease identification using transfer learning and convolutional neural networks. *MDPI Electronics*, 9(6), 937. https://doi.org/10.3390/electronics9060937

Schneider, J.; Börner, D.; Van Rosmalen, P.; & Specht, M. (2015). Presentation trainer, your public speaking multimodal coach. *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction - ICMI'15*. 539-546. doi:10.1145/2818346.2830603

Shu, Z.; Wang, P. & Zhan, W. (2020). The research and implementation of human posture recognition algorithm via OpenPose. *2nd International Conference on Artificial Intelligence and Advanced Manufacture (AIAM)*. doi:10.1109/aiam50918.2020.00023

Terhürne, P.; Schwartz, B.; Baur, T.; Schiller, D.; Eberhardt, S. T.; André, E., & Lutz, W. (2022). Validation and application of the non-verbal behavior analyzer: An automated tool to assess non-verbal emotional expressions in psychotherapy. *Frontiers in Psychiatry, 13*. https://doi.org/10.3389/fpsyt.2022.1026015

TorchVision: sus mantenedores y contribuyentes. (2016). *TorchVision: PyTorch's Computer Vision library*. [Repositorio GitHub]. Disponible en: https://github.com/pytorch/vision

Verhulst, B.; Lodge, M., & Lavine, H. (2010). The attractiveness halo: Why some candidates are perceived more favorably than others. *Journal of Nonverbal Behavior, 34*(2), 111–117. https://doi.org/10.1007/s10919-009-0084-z

Vinciarelli, A.,; Pantic, M., & Bourlard, H. (2009). Social signal processing: Survey of an emerging domain. *Image and Vision Computing*, 27(12), 1743-1759. https://doi.org/10.1016/j.imavis.2008.11.007

Virtanen, P., et al. (2020). SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nature Methods*, 17, 261–272. https://doi.org/10.1038/s41592-019-0686-2

Wörtwein, T.; Chollet, M.; Schauerte, B.; Morency, L.; Stiefelhagen, R. & Scherer, S. (2015). Multimodal Ppublic speaking performance assessment. *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction - ICMI'15*. doi:10.1145/2818346.2820762

Wu, M. (2024). Gesture recognition based on deep learning: A Review. *EAI Endorsed Transactions on E-Learning*, *10*. https://doi.org/10.4108/eetel.5191

Yang, H. (2014) Sign language recognition with the Kinect sensor based on conditional random fields. *Sensors (Basel)*, 15(1), 135-47. doi: 10.3390/s150100135.

Zheng, C.; Wu, W.; Chen, C.; Yang, T.; Zhu, S.; Shen, J.; Kehtarnavaz, N., & Shah, M. (2023). Deep learning-based human pose estimation: A survey. *ACM Computing Surveys, 56*(1), Article 11, 37 pages. https://doi.org/10.1145/3603618

Zhu, Y.; Yang, Z. & Yuan, B. (2013). Reconocimiento de gestos de la mano basado en la visión, *Conferencia internacional sobre ciencias de los servicios (ICSS)* de 2013, Shenzhen, China, 260-265, doi: 10.1109/ICSS.2013.40.

Anexos

A. Consentimiento informado para la participación en el estudio

A continuación, se muestra el consentimiento que se ha presentado a los participantes del proyecto.

CONSENTIMIENTO INFORMADO PARA LA PARTICIPACIÓN EN EL ESTUDIO

Título del estudio: Hacia un sistema de reconocimiento del lenguaje corporal en presentaciones orales utilizando técnicas de *machine learning*

Investigadora Principal: Yulieth Hoyos Vera

Tutor: Ioannis Dimitriadis Damoulis

Institución: Universidad de Valladolid, Escuela Técnica Superior de Ingenieros de Telecomunicación

Estimado/a participante,

Le invitamos a participar en el estudio mencionado anteriormente, cuyo objetivo es desarrollar y evaluar un sistema automatizado para la identificación y mejora del lenguaje corporal en presentaciones orales mediante técnicas de aprendizaje automático (machine learning). Antes de decidir participar, es importante que comprenda el propósito del estudio, el procedimiento que se seguirá, y los posibles riesgos y beneficios asociados. Por favor, lea la siguiente información detenidamente y no dude en hacer cualquier pregunta que considere necesaria.

El propósito de este estudio es analizar y mejorar las competencias de comunicación oral, específicamente a través de la evaluación del lenguaje corporal. Para ello, se utilizarán técnicas de machine learning para detectar y analizar posturas y gestos durante presentaciones orales, proporcionando retroalimentación cuantitativa sobre áreas de mejora.

Como participante, se le pedirá que realice una presentación oral de aproximadamente 5 a 10 minutos de duración en un entorno controlado, durante la cual se grabarán en video sus movimientos y posturas. Se le indicará realizar ciertas posturas y gestos específicos para evaluar la variabilidad de su lenguaje corporal. Estos datos serán procesados y analizados utilizando herramientas de visión por computadora para identificar patrones en su comunicación no verbal.

Todos los datos recogidos durante el estudio serán tratados de manera confidencial y anónima. Sus grabaciones de video y cualquier otra información identificativa serán codificadas y almacenadas de manera segura para garantizar su privacidad. Solo los investigadores autorizados tendrán acceso a los datos sin codificar. Los resultados del estudio serán presentados de manera agregada y no incluirán información que pueda identificarle personalmente. Además, los datos de las grabaciones de video serán utilizados únicamente para los fines de este estudio y no se compartirán con terceros sin su consentimiento explícito.

Su participación en este estudio es completamente voluntaria. Usted tiene el derecho de negarse a participar o de retirarse en cualquier momento sin necesidad de proporcionar ninguna explicación y sin que ello implique ninguna penalización o pérdida de beneficios a los que de otro modo tenga derecho. Si decide retirarse, todos los datos recopilados de su participación hasta ese momento serán eliminados de nuestra base de datos.

No se anticipan riesgos significativos derivados de su participación en este estudio. Sin embargo, algunas personas pueden experimentar una ligera incomodidad debido a la grabación en video de sus presentaciones. Los beneficios potenciales de participar incluyen la obtención de una retroalimentación detallada sobre sus habilidades de presentación y la oportunidad de mejorar sus competencias de comunicación oral.

Este estudio ha sido revisado y aprobado por el Comité de Ética de Investigación de la Universidad de Valladolid. Nos comprometemos a seguir todas las normas éticas y legales para la protección de su privacidad y bienestar durante todo el estudio. Si tiene alguna pregunta o necesita más información sobre este estudio, no dude en ponerse en contacto con la investigadora principal.

He leído y comprendido la información proporcionada más arriba. He tenido la oportunidad de hacer preguntas y todas mis preguntas han sido respondidas satisfactoriamente. Entiendo que mi participación es voluntaria y que puedo retirarme del estudio en cualquier momento sin ninguna penalización. Al firmar este documento, consiento participar en este estudio de investigación.

Nombre del participante (en letra de imprenta): _____

Firma del participante: _____

Fecha: _____

Nombre del investigador (en letra de imprenta): _____

Firma del investigador: _____

Fecha: _____

Agradecemos sinceramente su participación y colaboración en este estudio. Su contribución es fundamental para el avance del conocimiento en la evaluación del lenguaje corporal en presentaciones orales y en la mejora de las técnicas de comunicación en diversos contextos.

B. Código generado para el aumento de imágenes en el conjunto de datos

```python
#For Augmentation

import os
import glob
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.preprocessing.image import load_img, img_to_array, save_img

def augment_images_in_folder(input_folder, output_folder, num_augmented_images):
    # Create an instance of the ImageDataGenerator with augmentation parameters
    datagen = ImageDataGenerator(
        rotation_range=40,       #Random rotation between 0-40 degrees
        width_shift_range=0.2,   #Random horizontal shift
        height_shift_range=0.2,  #Random vertical shit
        shear_range=0.2,         #shearing transformation
        zoom_range=0.2,          #Random zoom
        horizontal_flip=True,    #Ramdon horizontal flipping
        fill_mode='nearest'      #Fill in the pixels after augmentation
    )

    # Get all images in the input folder (assuming typical image formats)
    image_paths = glob.glob(os.path.join(input_folder, '*.[jp][pn]g'))  # Change this pattern based on y
    total_images = len(image_paths)
    lesser_flag = True
    index = -1
    if total_images == 0:
        print("No images found in the folder.")
        return
    if total_images < num_augmented_images:
      lesser_flag = True
      augment_per_image = num_augmented_images // total_images
    elif total_images > num_augmented_images:
      index = num_augmented_images
      augment_per_image = 1
```

```
    def foof():
        count = 1
        for image_path in image_paths[:index]:
            if count > num_augmented_images:
                break

            # Load the image
            img = load_img(image_path)  # PIL image
            img_array = img_to_array(img)  # Convert to numpy array
            img_array = img_array.reshape((1,) + img_array.shape)  # Reshape for the generator

            # Extract the image file name and extension
            base_name = os.path.basename(image_path)
            file_name, file_ext = os.path.splitext(base_name)

            # Generate augmented images and save them
            for batch in datagen.flow(img_array, batch_size=1):
                if count > num_augmented_images:
                    break
                # Save the augmented image with the specified name convention
                save_img(os.path.join(output_folder, f"{file_name}_AUGMENTED_{count}{file_ext}"), batch[0])
                count += 1


                if count > augment_per_image:
                    break
            count = 0


        print(f"Generated {num_augmented_images} augmented images in {output_folder}.")
    foof()

# Example usage:
# augment_images_in_folder('train/positive/', 'train/positive/', num_augmented_images=3000)


augment_images_in_folder('train/positive/','train/positive/', num_augmented_images=1050)


augment_images_in_folder('train/negative/','train/negative/', num_augmented_images=1050)


augment_images_in_folder('test/positive/','test/positive/', num_augmented_images=450)


augment_images_in_folder('test/negative/','test/negative/', num_augmented_images=450)
```

```
Generated 1050 augmented images in train/positive/.
Generated 1050 augmented images in train/negative/.
Generated 450 augmented images in test/positive/.
Generated 450 augmented images in test/negative/.
```

C. Código de entrenamiento y evaluación del modelo

A continuación, se presenta el código del *notebook* con el cual se entrenó el modelo.

training the classification model

```
[ ] %cd /content/drive/MyDrive/DATASET/DATASET
```

```
/content/drive/MyDrive/DATASET/DATASET
```

```
ls
```

```
code/                    main.py      Pose.drawio.png       test/
confusion_matrix.png     Models/      PoseEstimation-1.pdf  train/
'Copia de pose_model.h5'  output.jpg  PoseEstimation.ipynb  VID_20240730_082933.mp4
```

+ Code + Text

```
[ ] import tensorflow as tf
    # Import TensorFlow and Keras modules for deep learning tasks

    # Import concatenate function for merging tensors along a specified axis
    from tensorflow.keras.layers import concatenate

    # Import Input layer for defining the input tensor of the model
    # Note: For TensorFlow 2.x, it's recommended to import Input from tensorflow.keras.layers
    from keras.layers import Input

    # Import the Adam optimizer for adaptive learning rate optimization
    from tensorflow.keras.optimizers import Adam

    # Import ImageDataGenerator for real-time data augmentation and image preprocessing
    from tensorflow.keras.preprocessing.image import ImageDataGenerator

    # Import common Keras layers used in building convolutional neural networks
    from tensorflow.keras.layers import MaxPooling2D, Flatten, Conv2D, Dense, BatchNormalization, GlobalAveragePooling2D, Dropout

    # Import DenseNet169 model, a pre-trained deep convolutional neural network
    from tensorflow.keras.applications.densenet import DenseNet169

    # Import MobileNetV2 model, a lightweight convolutional neural network designed for mobile and edge devices
    from tensorflow.keras.applications.mobilenet_v2 import MobileNetV2
```

```
[ ] # Define the main directory where the dataset is stored
    main_dir = "/content/drive/MyDrive/DATASET/DATASET/"

    # Define the subdirectories for different data splits within the main directory
    train_data_dir = main_dir + "train/Train"               # Directory containing training data
    test_data_dir = main_dir + "test/Test"                  # Directory containing test data

    # Define subdirectories within the training data directory for positive and negative samples
    train_n = train_data_dir + 'bg_positive/'           # Directory for positive class images in the training set
    train_p = train_data_dir + 'bg_negative/'           # Directory for negative class images in the training set
```

```
[ ] # Define the main directory where the dataset is stored
    main_dir = "/content/drive/MyDrive/DATASET/DATASET/"

    # Define the subdirectories for different data splits within the main directory
    train_data_dir = main_dir + "train/Train"               # Directory containing training data
    test_data_dir = main_dir + "test/Test"                  # Directory containing test data

    # Define subdirectories within the training data directory for positive and negative samples
    train_n = train_data_dir + 'bg_positive/'           # Directory for positive class images in the training set
    train_p = train_data_dir + 'bg_negative/'           # Directory for negative class images in the training set
```

```
[ ] train_datagen = ImageDataGenerator(rescale=1. / 255)
    test_datagen = ImageDataGenerator(rescale=1. / 255)

    # Create an instance of ImageDataGenerator for training data
    # Rescales pixel values to the range [0, 1] by dividing by 255
    train_datagen = ImageDataGenerator(rescale=1. / 255)

    # Create an instance of ImageDataGenerator for testing data
    # Rescales pixel values to the range [0, 1] by dividing by 255
    test_datagen = ImageDataGenerator(rescale=1. / 255)

    # Rescales pixel values to the range [0, 1] by dividing by 255
    # (Currently commented out, but can be used similarly to train_datagen and test_datagen)
```

```python
import sys
from IPython.display import display, Javascript

def suppress_stdout():
    with open(os.devnull, "w") as devnull:
        old_stdout = sys.stdout
        sys.stdout = devnull
        try:
            yield
        finally:
            sys.stdout = old_stdout
# Define image dimensions and batch size
img_width, img_height = [224, 224]  # Set the width and height for resizing images
batch_size = 15  # Number of images to be processed in a single batch

# Create a data generator for the training set
train_generator = train_datagen.flow_from_directory(
    train_data_dir,                  # Directory where training data is stored
    target_size=(img_width, img_height),  # Resize images to the specified dimensions
    batch_size=batch_size,           # Number of images to return in each batch
    class_mode='categorical',        # Use categorical labels (one-hot encoding) for classification
    shuffle=True                     # Shuffle the data at the beginning of each epoch
)



# Create a data generator for the test set
test_generator = test_datagen.flow_from_directory(
    test_data_dir,                   # Directory where test data is stored
    target_size=(img_height, img_width),  # Resize images to the specified dimensions
    batch_size=batch_size,           # Number of images to return in each batch
    class_mode='categorical'   ,     # Use categorical labels (one-hot encoding) for classification
    shuffle=False
)
display(Javascript('Jupyter.notebook.clear_output()'))
```

```
Found 3787 images belonging to 2 classes.
Found 1410 images belonging to 2 classes.
```

```python
[ ] # Define the shape of the input images
    input_shape = (224, 224, 3)  # Height, width, and number of channels (RGB) of the input images

    # Create an input layer for the model
    input_layer = Input(shape=(224, 224, 3))  # Input layer that expects images of size 224x224 with 3 color channels (RGB)
```

Training

```python
# Load the MobileNetV2 model as the base model
# Pre-trained on ImageNet data, excluding the fully connected layers at the top (include_top=False)
mobilenet_base = MobileNetV2(weights='imagenet', input_shape=input_shape, include_top=False)

# Load the DenseNet169 model as the base model
# Pre-trained on ImageNet data, excluding the fully connected layers at the top (include_top=False)
densenet_base = DenseNet169(weights='imagenet', input_shape=input_shape, include_top=False)

# Freeze the layers of the MobileNetV2 model to prevent them from being updated during training
for layer in mobilenet_base.layers:
    layer.trainable = False

# Freeze the layers of the DenseNet169 model to prevent them from being updated during training
for layer in densenet_base.layers:
    layer.trainable = False

# Create the model using MobileNetV2 as the base
model_mobilenet = mobilenet_base(input_layer)  # Apply MobileNetV2 base model to the input layer

# Apply GlobalAveragePooling2D layer to reduce the dimensionality of the output from the base model
# This layer computes the average output of each feature map
model_mobilenet = GlobalAveragePooling2D()(model_mobilenet)

# Flatten the output of the GlobalAveragePooling2D layer to create a 1D tensor
# This is typically done before adding fully connected layers or classifiers
output_mobilenet = Flatten()(model_mobilenet)
```

```
Downloading data from https://storage.googleapis.com/tensorflow/keras-applications/mobilenet_v2/mobilenet_v2_weights_tf_dim_ordering_tf_kernels_1.0_224_no_top.h5
9406464/9406464 ━━━━━━━━━━━━━━━━━━━━ 1s 0us/step
Downloading data from https://storage.googleapis.com/tensorflow/keras-applications/densenet/densenet169_weights_tf_dim_ordering_tf_kernels_notop.h5
51877672/51877672 ━━━━━━━━━━━━━━━━━━━━ 2s 0us/step
```

```
[ ] # Create the model using DenseNet169 as the base
    model_densenet = densenet_base(input_layer)  # Apply DenseNet169 base model to the input layer

    # Apply GlobalAveragePooling2D layer to reduce the dimensionality of the output from the base model
    # This layer computes the average output of each feature map from DenseNet169
    model_densenet = GlobalAveragePooling2D()(model_densenet)

    # Flatten the output of the GlobalAveragePooling2D layer to create a 1D tensor
    # This is typically done before adding fully connected layers or classifiers
    output_densenet = Flatten()(model_densenet)

    # Concatenate the flattened outputs from both MobileNetV2 and DenseNet169
    # Merges the two feature vectors into one combined feature vector
    merged = tf.keras.layers.Concatenate()([output_mobilenet, output_densenet])
```

```
    # Apply Batch Normalization to the concatenated feature vector
    # Batch Normalization helps to stabilize and accelerate the training by normalizing the activations
    x = BatchNormalization()(merged)

    # Add a Dense layer with 256 units and ReLU activation
    # This layer learns a transformation of the input with 256 hidden units and applies the ReLU activation function
    x = Dense(256, activation='relu')(x)

    # Apply Dropout with a rate of 0.1
    # Dropout is a regularization technique that randomly sets a fraction of input units to zero during training to prevent overfitting
    x = Dropout(0.1)(x)

    # Apply another Batch Normalization
    # Normalizes the activations after applying Dropout
    x = BatchNormalization()(x)

    # Add another Dense layer with 128 units and ReLU activation
    # This layer further processes the output from the previous layer with 128 hidden units and the ReLU activation function
    x = Dense(128, activation='relu')(x)

    # Apply Dropout again with a rate of 0.5
    # Dropout is applied to the output of the previous Dense layer to prevent overfitting
    x = Dropout(0.1)(x)

    # Add the final Dense layer with 2 units and softmax activation
    # The output layer uses softmax activation to produce a probability distribution over 2 classes (e.g., negative and positive)
    x = Dense(2, activation='softmax')(x)

    # Define the final model
    # The model takes the input from the input_layer and outputs the result from the last Dense layer
    stacked_model = tf.keras.models.Model(inputs=input_layer, outputs=x)
```

```
# Define the Adam optimizer with a learning rate of 0.0001
# Adam is an adaptive learning rate optimization algorithm that combines the benefits of two other extensions of stochastic gradient descent
optm = Adam(learning_rate=0.0001)

# Compile the model with the specified loss function, optimizer, and metrics
stacked_model.compile(
    loss='binary_crossentropy',    # Loss function used for binary classification problems
    optimizer=optm,                # Optimizer to update model weights
    metrics=['accuracy']           # Metric to evaluate the model performance
)
```

```
# Train the stacked model with the specified training parameters and callbacks
stacked_history = stacked_model.fit(
    train_generator,                    # Training data generator providing batches of images and labels for training
    epochs=100,                         # Number of epochs to train the model; each epoch involves one full pass through the training data
    batch_size=15,                      # Batch size to be used during training; determines the number of samples processed before the model's weights are updated
    validation_data=test_generator,   # Validation data generator providing batches of images and labels for validation
)
stacked_model.save('pose_model.h5')
```

Testing

```
[ ] %cd /content/drive/MyDrive//DATASET/DATASET
```

```
/content/drive/MyDrive/DATASET/DATASET
```

```
[ ] from tensorflow.keras.models import load_model

    # Load the previously saved model from the specified file path
    model = load_model('Models/pose_model.h5')
```

```
WARNING:absl:Compiled the loaded model, but the compiled metrics have yet to be built. `model.compile_metrics` will be empty until you train or evaluate the model.
```

```
# Evaluate the loaded model on the test data
test_loss, test_acc = model.evaluate(
    test_generator,  # Data generator providing batches of test images and labels
    steps=len(test_generator)  # Number of steps (batches) to draw from the generator for evaluation
)

# Print the accuracy of the model on the test dataset
print(f'Test Accuracy: {test_acc}')

# Print the loss of the model on the test dataset
print(f'Test Loss: {test_loss}')
```

```
89/89 ━━━━━━━━━━━━━━━━━━━━ 36s 400ms/step - accuracy: 0.8590 - loss: 0.4134
Test Accuracy: 0.8588652610778809
Test Loss: 0.4464718699455261
```

D. Encuesta para evaluar la facilidad de uso de la interfaz

A continuación, se presenta la encuesta utilizada para evaluar la interfaz

Encuesta para Evaluar la Interfaz

Instrucciones:
Por favor, responde las siguientes preguntas sobre tu experiencia al utilizar la interfaz. Tus respuestas son esenciales para ayudarnos a mejorar la usabilidad del sistema.

1. Edad:

- 18-25
- 26-35
- 36-45
- 46 o más

2. Género:

- Femenino
- Masculino
- Prefiero no decir
- Otro (especificar): _____

3. Experiencia previa con interfaces similares:

- Ninguna
- Básica
- Moderada
- Avanzada

4. Interacción con los elementos de la interfaz

4.1. ¿Te resultó fácil interactuar con los botones de la interfaz?

- 1 - Muy difícil
- 2 - Difícil
- 3 - Neutral
- 4 - Fácil
- 5 - Muy fácil

4.2. ¿Tuviste algún problema con la funcionalidad de algún elemento de la interfaz?

- Sí
- No
- Si respondiste "Sí", por favor explica:

---

5. Satisfacción General

5.1. ¿Qué tan satisfecho estás con la experiencia general de uso de la interfaz?

- 1 - Muy insatisfecho
- 2 - Insatisfecho
- 3 - Neutral
- 4 - Satisfecho
- 5 - Muy satisfecho