



Universidad Politécnica
de Madrid



**Escuela Técnica Superior de
Ingenieros Informáticos**

Máster Universitario en Software y Sistemas

Trabajo Fin de Máster

**Segmentación e Identificación de
Gestos en Imágenes y Vídeos: Aplicación
en la Traducción de la Lengua de Signos
Española (LSE)**

Autor(a): Abner García Ferrufino

Tutor: Raúl Alonso Calvo

Tutor: José Crespo del Arco

Madrid, julio 2024

Este Trabajo Fin de Máster se ha depositado en la ETSI Informáticos de la Universidad Politécnica de Madrid para su defensa.

Trabajo Fin de Máster

Máster Universitario en Software y Sistemas

Título: Segmentación e Identificación de Gestos en Imágenes y Videos:
Aplicación en la Traducción de la Lengua de Signos Española (LSE)

Junio 2024

Autor: Abner García Ferrufino

Tutores:

Raúl Alonso Calvo

Departamento de Lenguajes, Sistemas Informáticos e Ingeniería de Software

José Crespo del Arco

Departamento de Lenguajes, Sistemas Informáticos e Ingeniería de Software

ETSI Informáticos

Universidad Politécnica de Madrid

Agradecimientos

Este trabajo, el cual representa la culminación de mis estudios del Máster Universitario en Software y Sistemas, no hubiera sido posible de no haber sido por el apoyo incondicional de mi familia. Agradezco a mis padres, Telma Ferrufino Reyes y José García Meléndez y a mi hermana Kathia García Ferrufino por ser una parte íntegra en mi formación personal y académica.

Agradezco a la Fundación Carolina, quienes a través del programa “Becas Bicentenario” me brindaron la oportunidad de cumplir mi deseo de realizar mis estudios de postgrado en el exterior y de poder vivir de primera mano la experiencia de estar en un país con costumbres y cultura diferente. A pesar de tener aspectos en común entre España y Honduras como el idioma, haber vivido durante este periodo académico a Madrid es una de las experiencias más valiosas en mi vida académica y personal.

Agradezco a la Universidad Politécnica de Madrid por aceptarme como parte de sus estudiantes para poder llegar a cabo este máster. Especialmente a mis tutores de trabajo de fin de máster, Raúl Alonso Calvo y José Crespo del Arco, quienes se dedicaron a dar revisión de mi trabajo durante este último semestre para lograr presentar un trabajo que esté a la altura de las expectativas de la universidad.

Finalmente, agradezco a mis amigos, a mis compañeros de máster, a los becarios de la Fundación Carolina y a todas las personas que tuve el honor de conocer durante mi estancia en esta ciudad. Finalizo este máster no solamente con más conocimientos en temas de informática, sino con una percepción distinta del mundo gracias a la gran diversidad cultural que nos caracteriza a cada uno de nosotros y que tuve el placer de observar de primera mano. Mis mejores deseos para cada uno de ustedes.

Resumen

Este estudio tiene como objetivo desarrollar un sistema de reconocimiento de gestos para la lengua de signos española para mejorar la comunicación entre personas sordas y oyentes. La investigación se enmarca en un contexto histórico que sitúa la formación de asociaciones de sordos en España a principios del siglo XX, en particular la Confederación Estatal de Personas Sordas (CNSE), al igual que la promulgación de la Ley 27/2007, de 23 de octubre, por la que se reconocen las lenguas de signos españolas y se regulan los medios de apoyo a la comunicación oral de las personas sordas, con discapacidad auditiva y sordociegas.

Motivado por la necesidad actual de cerrar las brechas de comunicación y promover la conciencia de la comunicación gestual, en este estudio se busca crear un prototipo de sistema que utilice técnicas avanzadas de aprendizaje automático y computación visual para identificar e interpretar gestos de lengua de signos española en texto. Los objetivos específicos incluyen establecer el estado del arte en la detección de gestos, generar un conjunto de datos de vocabulario en la lengua de signos española y analizar distintos modelos de aprendizaje automático para la detección de gestos en tiempo real.

Asimismo, se hace una revisión exhaustiva del estado del arte en el reconocimiento de gestos en lengua de signos, señalando avances significativos en el campo, en particular el crecimiento generalizado de la investigación y la aplicación en países como India, China y Estados Unidos. Analiza las técnicas de computación visual para la detección de gestos, como las redes neuronales convolucionales (CNN) y las redes neuronales recurrentes (RNN), y herramientas prácticas como Google MediaPipe para detectar puntos clave del cuerpo. También se exploran los algoritmos para la clasificación de gestos, incluida la agrupación de K-medias, los modelos ocultos de Markov y las máquinas de vectores de soporte.

Por otra parte, se evalúan diversas metodologías para el reconocimiento de la lengua de signos, que van desde cámaras de vídeo convencionales y dispositivos de seguimiento de movimiento como Microsoft Kinect hasta sensores portátiles. El estudio enfatiza el enfoque holístico para capturar múltiples partes del cuerpo, como las manos, el torso y la cara, para mejorar la precisión del reconocimiento de gestos.

El análisis comparativo de los modelos, incluidos InceptionResNetV2, redes neuronales de memoria a corto plazo extendida, unidades recurrentes cerradas, entre otras. Destacando la precisión y eficiencia en las épocas de cada modelo.

Esta investigación no solo avanza en las capacidades técnicas de los sistemas de reconocimiento de gestos, sino que también se alinea con objetivos sociales más amplios, como los descritos en la Agenda 2030 para el Desarrollo Sostenible de las Naciones Unidas, que aboga por una educación inclusiva y equitativa de calidad y la reducción de las desigualdades dentro de los países y entre ellos.

Abstract

This study aims to develop a gesture recognition system for Spanish sign language to improve communication between deaf and hearing people. The research is framed in a historical context that places the formation of associations of the deaf in Spain at the beginning of the twentieth century, in particular the State Confederation of Deaf People (CNSE), as well as the enactment of Law 27/2007, of October 23, which recognizes Spanish sign languages and regulates the means of support for oral communication of deaf people. hearing impaired and deafblind.

Motivated by the current need to close communication gaps and promote awareness of gestural communication, this study seeks to create a prototype system that uses advanced machine learning and visual computing techniques to identify and interpret Spanish sign language gestures in text. Specific objectives include establishing the state of the art in gesture detection, generating a vocabulary dataset in Spanish sign language, and analyzing different machine learning models for real-time gesture detection.

It also makes an exhaustive review of the state of the art in the recognition of gestures in sign language, pointing out significant advances in the field, in particular the widespread growth of research and application in countries such as India, China and the United States. It discusses visual computing techniques for gesture detection, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), and practical tools such as Google MediaPipe for detecting key points on the body. Algorithms for gesture classification, including K-mean clustering, hidden Markov models, and support vector machines, are also explored.

On the other hand, various methodologies for sign language recognition are being evaluated, ranging from conventional video cameras and motion tracking devices such as Microsoft Kinect to portable sensors. The study emphasizes the holistic approach to capturing multiple body parts, such as the hands, torso, and face, to improve the accuracy of gesture recognition.

Comparative analysis of models, including InceptionResNetV2, extended short-term memory neural networks, closed recurrent units, among others. Highlighting the precision and efficiency in the eras of each model.

This research not only advances the technical capabilities of gesture recognition systems, but also aligns with broader societal goals, such as those outlined in the United Nations' 2030 Agenda for Sustainable Development, which advocates for inclusive and equitable quality education and the reduction of inequalities within and between countries.

Tabla de contenidos

1	Introducción.....	1
1.1	Antecedentes.....	2
1.2	Motivación del estudio.....	3
1.3	Objetivo.....	3
2	Estado del arte	4
2.1.1	Lengua de Signos.....	4
2.1.2	Lengua de Signos Española	5
2.2	Detección de Gestos en Lengua de Signos	6
2.3	Técnicas de Visión por Computador	8
2.4	Técnicas de Reconocimiento de Lengua de Signos	10
2.4.1	Detección Visual	11
2.4.1.1	Uso de Cámaras de Vídeo Convencionales para la Detección de Gestos	12
2.4.1.2	Uso de Dispositivos de Rastreo de Movimientos para la Detección de Gestos	16
2.4.1.3	Uso de técnicas invasivas para la detección de gestos.....	23
2.4.1.4	Uso de cámaras estereoscópicas para la detección de gestos.	27
2.5	Desafíos en la detección de lengua de signos.....	31
2.5.1	Complejidad de la lengua de signos	31
2.5.2	Generación del conjunto de datos	33
2.6	Técnicas para la detección de gestos en lengua de signos.....	39
2.6.1	Limitaciones de los dispositivos para la detección de gestos en lengua de signos	41
3	Desarrollo	43
3.1	Especificación de hardware utilizado.....	43
3.2	Elaboración del conjunto de datos.....	44
3.3	Implementación de solución	48
4	Resultados obtenidos	51
5	Conclusiones	53
5.1	Consecución de los objetivos planteados	53
5.2	Problemas encontrados	54
5.3	Conclusiones personales	54
5.4	Líneas futuras.....	55
6	Bibliografía	56
7	Anexos.....	61
7.1	Código del sistema desarrollado para la detección de gestos en lengua de signos española.....	61

1 Introducción

La lengua de signos o, como se conoce en otras regiones del mundo, lengua de señas, se define como una colección de gestos y expresiones cuyo propósito es el de comunicar palabras e ideas entre dos o más individuos. Este tipo de lengua es un aspecto común alrededor del mundo, se estima que más de 1.500 millones de personas tendrán algún tipo de disminución de la capacidad auditiva a lo largo del curso de su vida.

Debido a la cantidad de gente que requiere utilizar la lengua de signos para comunicarse, los gobiernos e instituciones internacionales centran sus esfuerzos en elaborar políticas y normas para mejorar la accesibilidad en los servicios ofrecidos por el estado, así como promover estándares que permitan a los proveedores privados facilitar la accesibilidad de los servicios prestados. En el caso específico de España, el gobierno promueve el uso de la lengua de signos española mediante la Ley 27/2007, de 23 de octubre, por la que se reconocen las lenguas de signos españolas y se regulan los medios de apoyo a la comunicación oral de las personas sordas, con discapacidad auditiva y sordociegas.

Estas políticas y normas de accesibilidad tienen un fuerte componente tecnológico, ya que en la sociedad actual el acceso a servicios digitales es imprescindible para las personas, independientemente de sus condiciones físicas o cognitivas. Debido a esto, múltiples estudios se han realizado en distintos países para promover la accesibilidad a través de tecnologías de asistencia en los servicios y recursos que existen en internet hoy en día.

Asimismo, con el auge de la inteligencia artificial y el aprendizaje automatizado, el reconocimiento de gestos en las lenguas de signos se está volviendo una tarea de relativa facilidad para los dispositivos actuales, por lo que el desarrollo de sistemas de detección de gestos ya existen actualmente en una amplia variedad de alternativas, desde dispositivos físicos como guantes para la detección de movimiento, hasta el uso de cámaras de vídeo para detectar el movimiento de las personas que expresan sus ideas mediante lengua de signos.

El presente estudio tiene como objetivo desarrollar el prototipo de un sistema de reconocimiento de gestos mediante técnicas de aprendizaje automatizado y computación visual para la traducción de la lengua de signos española a texto. Para lo cual se realizó un estudio exhaustivo de las técnicas que actualmente se utilizan para la detección de individuos en imágenes y el rastreo de movimiento en vídeos obtenidos desde una cámara web.

La presente memoria de Tesis de Fin de Máster está dividida en siete capítulos. El capítulo 2 describe el estado del arte de los sistemas de detección de gestos en lengua de signos, detallando las técnicas utilizadas para estos tipos de sistemas. El capítulo 3 se enfoca en el proceso mediante el cual se generó el conjunto de datos con el que se entrenó la red neuronal diseñada para este sistema y en el diseño de la red neuronal implementada en Python. En el capítulo 4 se discuten los resultados obtenidos por la red neuronal, haciendo un comparativo de la precisión obtenida con distintas configuraciones de entrenamiento. El capítulo 5 incluye la consecución de los objetivos planteados, las conclusiones del estudio y líneas futuras de estudio relacionados con este sistema. El capítulo 6 muestra las referencias utilizadas para el marco teórico de este estudio y, finalmente, el capítulo 7 incluye los anexos del estudio, específicamente el código desarrollado para el sistema objeto de este estudio.

1.1 Antecedentes

En las primeras décadas del siglo XX nacen las primeras asociaciones de sordos en España, siendo una de estas la Confederación Estatal de Personas Sordas (CNSE), cuyo propósito es el de “construir unidad, esta palabra, este símbolo que nos caracteriza, y que ha hecho posible que las personas sordas nos acogiéramos unas a otras y conformáramos las asociaciones y, que a su vez, éstas se unieran alrededor de la CNSE” (Confederación Estatal de Personas Sordas, s.f.).

Durante los años 90 surgieron una serie de movimientos que impulsarían la difusión de la lengua de signos española en el España. Específicamente en el año 1993 cuando el Ministerio de Educación y Cultura y la Confederación Estatal de Personas Sordas establecieron un convenio para el uso de intérpretes de lengua de signos en salones de clases y que, de acuerdo con Rodríguez y Roche (2007), haya ocasionado que la figura del intérprete de lengua de signos fuera “experimentado una evolución notable [...] porque este tipo de convenio se fue extendiendo a distintas autonomías y ello ha permitido que el número de intérpretes en los institutos de secundaria se haya ido incrementando”

El 25 de octubre de 2007 entra en vigor la Ley 27/2007, de 23 de octubre, por la que se reconocen las lenguas de signos españolas y se regulan los medios de apoyo a la comunicación oral de las personas sordas, con discapacidad auditiva y sordociegas. De acuerdo con el Centro de Normalización de Lingüística de la Lengua de Signos Española (2022), esta ley “ha servido para sistematizar [...] la respuesta de la sociedad a los ejes básicos de los derechos de las personas sordas, [...] transmite el compromiso del Estado [...] con la inclusión, la promoción efectiva de los derechos y la remoción de obstáculos”.

Por otra parte, en el ámbito de la inteligencia artificial y el aprendizaje automatizado, durante finales de la década de los 50 comienzan a darse los primeros diseños sobre modelos que simulan la conexión de las neuronas en el cerebro de los humanos. Es en esta época donde se introduce el concepto de “perceptrón”, el cual es la unidad fundamental de las redes neuronales utilizadas en el aprendizaje automatizado. Esto se convirtió en el prototipo de las redes neuronales artificiales modernas, y el modelo de su aprendizaje se acercó a los modelos de aprendizaje animal y humano desarrollados en psicología (Fradkov, 2020).

Con el avance del aprendizaje automatizado a lo largo del tiempo, nuevas técnicas se fueron aplicando en diversas áreas, como es la rama de estudio de la computación visual. Para finales de los años 90 se comenzaron a desarrollar las primeras propuestas de redes neuronales de memoria a corto plazo extendida, siendo estos de los modelos más utilizados para la detección de gestos en tiempo real.

Con la llegada del siglo XXI y la continua evolución de la capacidad de cómputo en los dispositivos electrónicos de uso diario de las personas, se fue facilitando el desarrollo de técnicas que permitieran detectar gestos en lengua de signos a través de estos dispositivos. Según Adeyanju, et al. (2021) con el reciente avance en el aprendizaje automático y los métodos de inteligencia computacional, los sistemas inteligentes de reconocimiento de lenguaje de señas continúan atrayendo la atención de investigadores académicos y profesionales industriales, especialmente en las últimas 2 décadas.

1.2 Motivación del estudio

La motivación de este estudio surge a partir del deseo de desarrollar un sistema de reconocimiento de gestos en lengua de signos para cerrar las brechas de comunicación entre las comunidades de personas sordas y personas oyentes. Asimismo, se desea promover el interés del público en general en este tipo de soluciones para dar a conocer la diversidad que se tiene al momento de querer comunicar ideas de formas gestuales.

Esta tecnología también tiene como objetivo facilitar interacciones más inclusivas y accesibles, empoderando a las personas que usan el lenguaje de signos para comunicarse de manera más efectiva con aquellos que no dominan esta lengua. Por ejemplo, mediante la creación de un homólogo a la funcionalidad de subtítulos en tiempo real que cuentan muchos reproductores de vídeo online, en el cual se identifiquen los gestos que un individuo esté realizando y puedan ser mostrados en texto legible en la pantalla para que cualquier persona que esté viendo el vídeo pueda entender las ideas que el signante está expresando.

De igual forma, la implementación de sistemas de reconocimiento de gestos en lengua de signos puede abrir oportunidades en entornos como las redes sociales, en las que una persona cuyo medio de comunicación principal es la lengua de signos pueda interactuar con facilidad

Además, un sistema de este tipo puede mejorar las oportunidades educativas, proporcionar un mejor acceso a los servicios y promover la integración social de la comunidad de personas sordas. Al aprovechar tecnologías avanzadas como el aprendizaje automático y la computación visual, el sistema de reconocimiento de gestos de lengua de signos tiene el potencial de transformar la comunicación, fomentando una sociedad más inclusiva donde se minimizan las barreras lingüísticas.

1.3 Objetivo

El presente estudio tiene como objetivo general desarrollar el prototipo de un sistema de reconocimiento de gestos mediante técnicas de aprendizaje automatizado y computación visual para la traducción de la lengua de signos española a texto.

Para alcanzar de forma satisfactoria este objetivo, se requiere de enfocar el estudio en los siguientes objetivos específicos:

- a. Establecer el estado del arte en las técnicas de detección de gestos en lengua de signos a través de modelos de aprendizaje automatizado.
- b. Generar un conjunto de datos con una lista de expresiones y palabras que comprendan un subconjunto del vocabulario en la lengua de signos española.
- c. Analizar los distintos modelos de aprendizaje automático que puedan utilizarse para la detección de gestos en lengua de signos en tiempo real, para ser implementados en el sistema propuesto en este estudio.

2 Estado del arte

En la presente sección se describe en detalle el contexto de la lengua de signos, de la visión computacional y de los sistemas existentes hasta la fecha para la segmentación e identificación de gestos a través de imágenes y vídeos.

2.1.1 Lengua de Signos

La lengua de signos (también conocida como lengua de señas) puede definirse de forma general como una colección de gestos y expresiones cuyo propósito es el de comunicar palabras e ideas entre dos o más individuos. Desde una perspectiva formal, se puede definir la lengua de signos como una “lengua natural de expresión y configuración gesto-espacial y percepción visual [...] se basa en movimientos y expresiones a través de las manos, los ojos, el rostro, la boca y el cuerpo” (Vercher, 2018).

Por otra parte, se define la lengua de signos como “lengua natural de carácter visual, gestual y espacial con gramática propia que reúne todas las características y cumple las mismas funciones que cualquier otra lengua” (Confederación Estatal de Personas Sordas, s.f.).

La lengua de signos ha permitido la inclusión de un número considerable de personas que carecen parcial o totalmente la habilidad de escuchar. De acuerdo con los últimos reportes, “más de 1.500 millones de personas tendrán algún tipo de disminución de la capacidad auditiva a lo largo del curso de su vida, de las cuales al menos 430 millones necesitarán atención” (Organización Panamericana de la Salud, 2021).

Asimismo, es importante resaltar que existe una amplia gama de lenguas de signos alrededor del mundo, “no hay una única lengua de signos en el mundo, cada país tiene una o varias lenguas de signos que han evolucionado en el seno de sus comunidades lingüísticas, con independencia de las lenguas orales” (Confederación Estatal de Personas Sordas, s.f.).

Asimismo, la lengua de signos constituye un aspecto importante a tomar en cuenta para el desarrollo sostenible de la sociedad a futuro. La Organización de las Naciones Unidas (ONU), mediante la Agenda 2030 de la ONU para el Desarrollo Sostenible, busca favorecer a la sociedad en general, incluyendo a aquellos individuos que utilizan la lengua de signos como medio principal de comunicación. La agenda antes mencionada “es un plan de acción en favor de las personas, el planeta y la prosperidad. También tiene por objeto fortalecer la paz universal dentro de un concepto más amplio de la libertad” (ONU, 2015).

Dentro de la agenda de desarrollo sostenible para el 2030 de la ONU se encuentran una serie de objetivos que están en línea con la promoción de la lengua de signos en la sociedad, en la Tabla 1 se describe cada uno de los objetivos que, de forma directa o indirecta, involucran a las personas que utilizan lengua de signos.

No. Objetivo	Descripción
4.	Educación de calidad
4.5.	Eliminar las disparidades de género en la educación y asegurar el acceso igualitario a todos los niveles de la enseñanza y la formación profesional para las personas vulnerables, incluidas las personas con discapacidad, los pueblos indígenas y los niños en situaciones de vulnerabilidad.

No. Objetivo	Descripción
4.7.a.	Construir y adecuar instalaciones educativas que tengan en cuenta las necesidades de los niños y las personas con discapacidad y las diferencias de género, y que ofrezcan entornos de aprendizaje seguros, no violentos, inclusivos y eficaces para todos.
8.	Trabajo decente y crecimiento económico
8.5.	Lograr el empleo pleno y productivo y el trabajo decente para todas las mujeres y los hombres, incluidos los jóvenes y las personas con discapacidad, así como la igualdad de remuneración por trabajo de igual valor.
10.	Reducción de las Desigualdades
10.2.	Potenciar y promover la inclusión social, económica y política de todas las personas, independientemente de su edad, sexo, discapacidad, raza, etnia, origen, religión o situación económica u otra condición.
10.3.	Garantizar la igualdad de oportunidades y reducir la desigualdad de resultados, incluso eliminando las leyes, políticas y prácticas discriminatorias y promoviendo legislaciones, políticas y medidas adecuadas a ese respecto.

Tabla 1 Lista de objetivos de desarrollo sostenible establecidos por la ONU como parte de su agenda para el año 2030 en el que promueven la inclusión y accesibilidad en las personas, esto incluye a todas aquellas personas que utilizan la lengua de signos como medio de comunicación, tomado de ONU (2015).

La implementación de un sistema de detección de gestos en lengua de signos puede cerrar la brecha de comunicación entre individuos, promover la inclusión y mejorar la calidad de vida de las personas sordas y con problemas de audición al hacer que varios aspectos de la vida diaria sean más accesibles e integrados, en concordancia con los objetivos de desarrollo sostenible establecidos por la ONU.

2.1.2 Lengua de Signos Española

La Lengua de Signos Española se reconoce como lengua de signos oficial en el país y se regulan los medios de apoyo a la comunicación oral de las personas sordas, con discapacidad auditiva y sordociegas mediante la Ley 27/2007 de 23 de octubre de 2007.

La presente Ley tiene por objeto reconocer y regular la lengua de signos española como lengua de las personas sordas, con discapacidad auditiva y sordociegas en España que libremente decidan utilizarla, sin perjuicio del reconocimiento de la lengua de signos catalana en su ámbito de uso lingüístico, así como la regulación de los medios de apoyo a la comunicación oral (Gobierno de España, 2007).

En esta ley, se define a la lengua de signos como “sistemas lingüísticos de carácter visual, espacial, gestual y manual en cuya conformación intervienen factores históricos, culturales, lingüísticos y sociales, utilizadas tradicionalmente como lenguas por las personas sordas, con discapacidad auditiva y sordociegas signantes en España” (Gobierno de España, 2007).

De igual forma, dentro del territorio español existen múltiples lenguas de signos correspondientes a las comunidades autónomas del país. Entre las regiones que cuentan con una lengua de signos en España se encuentran Cataluña, Andalucía, Aragón, Canarias, Castilla y León, Comunidad Valenciana, Extremadura e Islas Baleares. En el caso específico de la lengua de signos catalana, está regulada por la misma Ley 27/2007 y por la Ley Catalana 17/2010; mientras que la lengua de signos andaluza se regula mediante la Ley Andaluza 11/2011.

La lengua de signos española tiene un valor cultural importante a nivel nacional, “cualquier vocablo de una lengua oral puede transmitirse mediante una lengua de modalidad viso-gestual” (Valdés-González & Martín-Antón, 2020). Por otra parte, el Consejo de Patrimonio Histórico del Ministerio de Cultura adscrito al Gobierno de España acordó en el año 2018 que la lengua de signos española, así como sus manifestaciones culturales, sean declaradas parte del Patrimonio Cultural Inmaterial.

Tomando en cuenta los objetivos de desarrollo sostenible de la ONU, el gobierno español, a través del Ministerio de Derechos Sociales, Consumo y Agenda 2030 ha realizado esfuerzos por contribuir en la medida de lo posible al cumplimiento de todos los objetivos dispuestos por la ONU. La emisión de las leyes antes mencionadas muestra un claro avance en la persecución de estos objetivos a largo plazo.

Asimismo, y en seguimiento a los objetivos de desarrollo sostenibles de la ONU, el gobierno español ha implementado su estrategia nacional sobre discapacidad para los años 2022 a 2030 para el acceso, goce y disfrute de los derechos humanos de las personas con discapacidad. De acuerdo con esta agenda, uno de los retos estratégicos es el de “Avanzar en asegurar el acceso a la información de las personas con discapacidad en todos los medios públicos de comunicación, [...] facilitando el uso de lengua de signos, [...] así como cualquier otro medio, modo o formato de comunicación accesible” (Ministerio de Derechos Sociales y Agenda 2030, 2022).

2.2 Detección de Gestos en Lengua de Signos

Actualmente existen diversas técnicas para el reconocimiento de signos, mismas que se discutirán en las secciones 2.2 y 2.3, haciendo énfasis en las técnicas de visión por computador más importantes utilizadas para este propósito y cómo estas técnicas son utilizadas para la detección de gestos en lengua de signos.

Asimismo, se tiene una amplia gama de técnicas utilizando aprendizaje automatizado para el reconocimiento de gestos. En un estudio elaborado por Adeyanju, et al. (2021) se llevó a cabo una recopilación bibliográfica de artículos relacionados a la detección de gestos mediante técnicas de aprendizaje automatizado, teniendo como resultado una distribución de los países con mayor cantidad de estudios en esta materia, la cual se muestra en la Tabla 2; siendo el top 3 India, China y los Estados Unidos de América.

Rango	País	Total de publicaciones	Total de citas referenciadas
1	India	123	2118
2	China	111	1353
3	Estados Unidos de América	70	3872

Rango	País	Total de publicaciones	Total de citas referenciadas
4	Reino Unido	29	632
5	Egipto	26	178
6	Arabia Saudita	24	169
7	Japón	23	316
8	Corea del Sur	19	253
9	España	19	360
10	Brasil	18	243
11	Canadá	17	328
12	Irán	17	186
13	Malasia	17	203
14	Taiwán	17	588
15	Francia	16	288
16	Italia	16	610

Tabla 2 Top 16 de instituciones académicas y nacionales con mayor cantidad de estudios referentes a la detección de gestos mediante aprendizaje automatizado en cada ubicación geográfica entre los años 2001 y 2021 (Adeyanju, et al., 2021).

Esta distribución en el desarrollo de las técnicas de reconocimiento de gestos para lengua de signos se muestra de forma más clara en la Ilustración 1, en la cual se incluye la distribución geográfica de los estudios representados en un mapa.

Países con Mayor Cantidad de Estudios Sobre Detección de Gestos Mediante Aprendizaje Automatizado
2001-2021

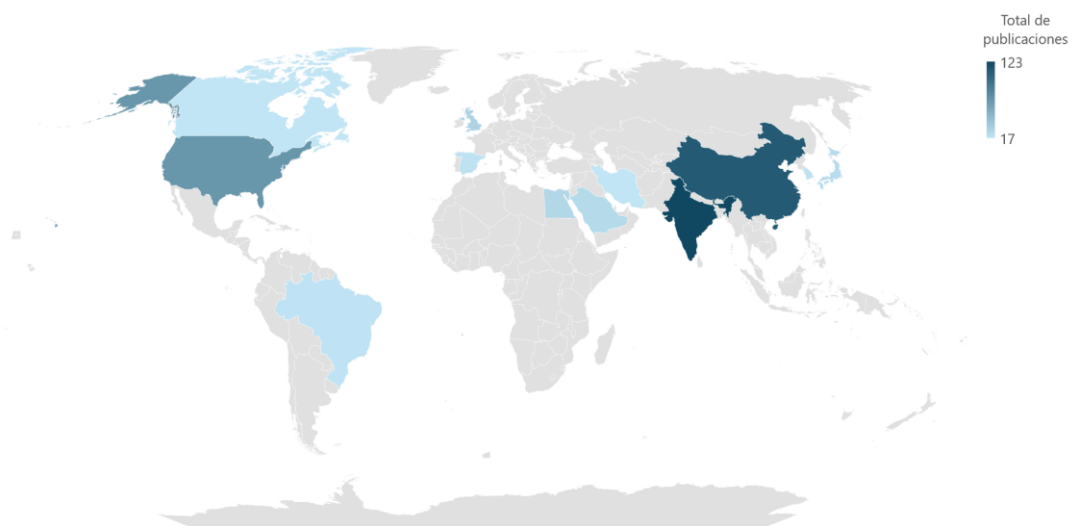


Ilustración 1 Países con mayor cantidad de estudios sobre detección de gestos mediante aprendizaje automatizado

2.3 Técnicas de Visión por Computador

Los métodos más comunes para la detección visual de gestos de lengua de signos son a través de algoritmos o sistemas de inteligencia artificial. Estos son sistemas de software o hardware que actúan en la dimensión física o digital percibiendo su entorno a través de la adquisición de datos, interpretando los datos recopilados, razonando sobre el conocimiento o procesando la información derivada de estos datos y decidiendo la o las mejores acciones a tomar para lograr el objetivo dado (European Commission, 2019).

Los sistemas de inteligencia artificial tienen una amplia gama de dominios y aplicaciones, ya que son sistemas que se pueden utilizar para solucionar una gran cantidad de problemas en entornos variados. La Comisión Europea (European Commission) define una taxonomía de los sistemas de inteligencia artificial basándose en tres aspectos:

- El uso del concepto de “agentes racionales” como entidades capaces de tomar decisiones basándose en su entorno.
- El avance del desarrollo de estas tecnologías en la industria.
- Otros aspectos destacables relacionados con la IA, pero no necesariamente relacionados con la tecnología, surgen como temas importantes en los documentos políticos y en el debate social.

La taxonomía propuesta por la Comisión Europea se describe en la tabla 1.

	Taxonomía de la Inteligencia Artificial (IA)	
	Dominio IA	Subdominio IA
Núcleo	Razonamiento	Representación del conocimiento
		Razonamiento automático
		Razonamiento del sentido común
	Planificación	Planificación y agenda
		Búsqueda
		Optimización
	Aprendizaje	Aprendizaje automático
	Comunicación	Procesamiento del lenguaje natural
	Percepción	Computación visual
		Procesamiento del audio
Transversal	Integración e interacción	Sistemas multi-agente
		Robótica y automatización
		Vehículos autónomos
	Servicios	Servicios de inteligencia artificial
	Ética y filosofía	Ética en la inteligencia artificial

Tabla 3 Dominios y subdominios de la inteligencia artificial (European Commission, 2020)

En el ámbito de la detección de lengua de signos, el subdominio del aprendizaje automatizado es el que predomina en las aplicaciones desarrolladas hasta el momento, este subdominio puede ser dividido en aprendizaje supervisado y no supervisado. El aprendizaje automatizado supervisado toma un conjunto de datos de entrenamiento conocidos y los utiliza para inferir un resultado a partir de datos previamente etiquetados, también conocido como “conjunto de datos de entrenamiento”, mientras que; el aprendizaje automatizado no supervisado se utiliza para extraer inferencias de conjuntos de datos con datos de entrada sin etiquetado previo (Adeyanju, et al., 2021).

En la Ilustración 2 se muestra en un gráfico de línea el crecimiento en el uso de algoritmos de aprendizaje automatizado para la detección de lengua de signos entre los años 2001 y 2021. Dicho gráfico muestra una clara tendencia al uso de esta tecnología para la detección de gestos en lengua de signos.

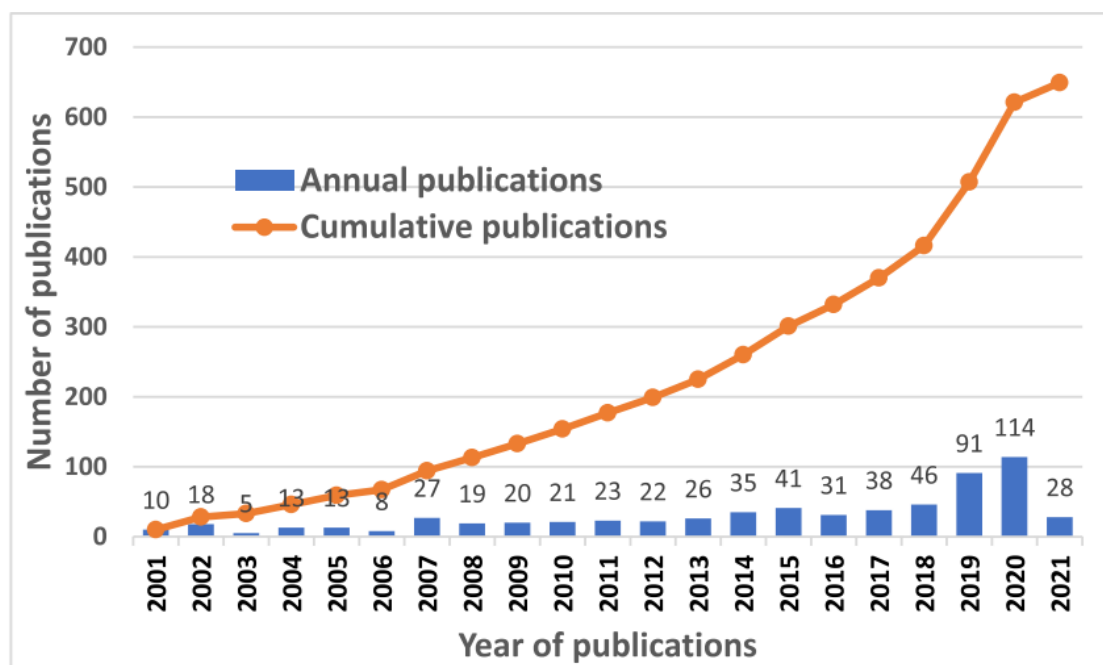


Ilustración 2 Distribución en el tiempo de las publicaciones anuales y acumulativas basadas en el reconocimiento mediante machine learning de la lengua de signos (Adeyanju, et al., 2021).

Usualmente cuando se trata el tema de aprendizaje automático se suele involucrar el concepto de redes neuronales. Para Chassagnon, et al. (2020) Las redes neuronales en el aprendizaje automático están diseñadas para imitar la forma en que el cerebro humano procesa la información, estas utilizan operaciones simples, que imitan la forma en que se comportan las neuronas para tratar la información.

Una de las ramas de la inteligencia artificial más utilizadas en este tipo de aplicaciones es el aprendizaje profundo (o deep learning). El aprendizaje profundo se refiere al uso de redes neuronales profundas, que son una configuración específica en la que las neuronas se organizan en múltiples capas sucesivas (LeCun, et al., 2015).

Gracias a los avances en la capacidad de cómputo de los ordenadores actuales, el aprendizaje profundo ha logrado tener una mayor adopción en los sistemas de inteligencia artificial actuales, incluyendo los sistemas de detección de gestos. Con la llegada de los enfoques de aprendizaje profundo, el área de reconocimiento de lenguaje de señas se ha enfrentado a una mejora significativa de la precisión en los últimos años (Rastgoo, et al., 2021).

El primer ejemplo del uso de aprendizaje profundo es mediante redes neuronales convolucionales. Las redes neuronales convolucionales se basan en la sucesión de operadores simples de convolución a diferentes escalas, siendo una convolución la inclusión de información de elementos agrupados mediante la aplicación de diferentes filtros que difieren de una capa a otra y su aplicación genera la entrada a la capa siguiente (Chassagnon, et al., 2020).

Al igual que las redes neuronales convolucionales, también existen las redes neuronales recurrentes, estas son una clase de redes neuronales diseñadas para procesar datos secuenciales mediante conexiones que forman ciclos dirigidos de información, lo que permite a este tipo de redes neuronales mantener una "memoria" de entradas anteriores. Las redes neuronales recurrentes son un tipo de red neuronal artificial que utiliza series temporales y datos secuenciales, estas se denominan recurrentes, ya que llevan a cabo la misma función para cada elemento de la secuencia, con la dependencia de los cálculos de estados anteriores Kothadiya, et al. (2022).

Dentro de las redes neuronales recurrentes más utilizadas para la detección de gestos, una de las más utilizadas es la red neuronal de memoria a corto plazo extendida unidireccional y bidireccional (también conocidas como BiLSTM o LSTM del inglés Bidirectional Long Short-Term Memory Neural Network y Long Short-Term Memory Neural Network, respectivamente). De acuerdo con Kumar et al. (2017), en los últimos años se ha demostrado que la red neuronal de memoria a corto plazo extendida bidireccional funciona mejor para aplicaciones de reconocimiento de voz y escritura a mano. Una particularidad de este tipo de redes neuronales es la forma en la que la información fluye, dado que cuenta con dos capas ocultas, una capa procesa la secuencia de entrada en dirección hacia adelante y la otra capa procesa la entrada en dirección en sentido contrario.

Por otra parte, uno de los algoritmos de aprendizaje no supervisado más populares es k-means. El algoritmo k-means es un método de agrupamiento no supervisado, no determinista, numérico e iterativo, cada grupo o clúster está representado por el valor medio de los objetos del clúster, luego se particiona un conjunto de n objetos en k cantidad clústeres para que la similitud entre clústeres sea baja y la similitud entre los clústeres sea alta; esta similitud se calcula con base en el valor medio de los objetos en un clúster (Yadav & Sharma, 2013).

Asimismo, se pueden utilizar otros modelos probabilísticos para la detección de gestos, como es el caso de los modelos de Markov, estos son unos modelos estadísticos utilizados para representar sistemas con estados ocultos, donde el sistema realiza transiciones entre estos estados a lo largo del tiempo. Mor, et al. (2020) indica que las transiciones de estado se refieren al cambio aleatorio en los estados del proceso de Markov en tiempo discreto. Por otra parte,

2.4 Técnicas de Reconocimiento de Lengua de Signos

El reconocimiento de lengua de signos se puede llevar a cabo de distintas formas, dependiendo de las herramientas y los acercamientos que se deseen utilizar, como ser sensores de movimiento, cámaras, entre otros. Hay cinco parámetros

principales en el lenguaje de signos, que son la forma de la mano, la orientación de la palma, el movimiento de las manos y brazos, la ubicación y las señales de expresión o gesticulaciones no manuales (Rastgoo, et al., 2021).

Los parámetros mencionados anteriormente representan gestos que forman parte integral de cualquier lengua de signos, ya que por definición la lengua de signos es una colección de gestos con un significado específico para las personas que se comunican de esta forma. Para reconocer efectivamente gestos en la lengua de En la Ilustración 3 se muestra la taxonomía de modelos profundos para el reconocimiento de lenguas de signos.

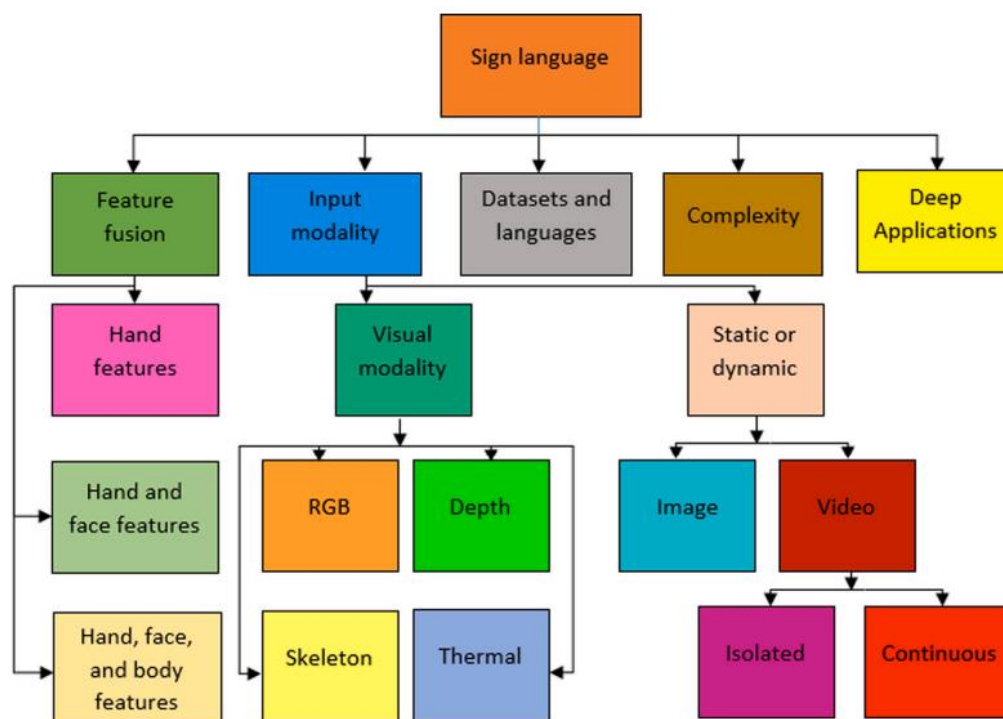


Ilustración 3 Taxonomía de modelos profundos para el reconocimiento de la lengua de signos (Rastgoo, et al., 2021)

Tomando en consideración únicamente las modalidades de entradas que de la Ilustración 3, se puede realizar la detección de lengua de signos mediante el uso de los valores RGB (rojo, verde y azul) de los píxeles o bien mediante la profundidad de objetos utilizados para la lengua de signos, esta detección puede realizarse tanto en imágenes como en videos.

Por otra parte, los sistemas de reconocimiento de signos pueden ser categorizados por la forma en que el sistema captura los gestos de las manos en lenguaje de señas. De acuerdo con Mohamed, et al. (2021), estos sistemas pueden ser basados en la visión o basados en dispositivos; los sistemas basados en visión capturan mediante cámaras los gestos elaborados, mientras que los sistemas basados en dispositivos emplean sensores conectados a los usuarios para obtener una medición de los gestos realizados.

2.4.1 Detección Visual

En la Ilustración 4 se muestran los distintos elementos que pueden ser identificados tanto en imágenes como en videos mediante modelos de clasificación (Rastgoo, et al., 2021). Dentro de este contexto, un modelo de clasificación puede definirse como un algoritmo informático que forman parte de la rama de la inteligencia artificial y que tienen como función principal la clasificación de datos basándose en un conjunto previo de información.

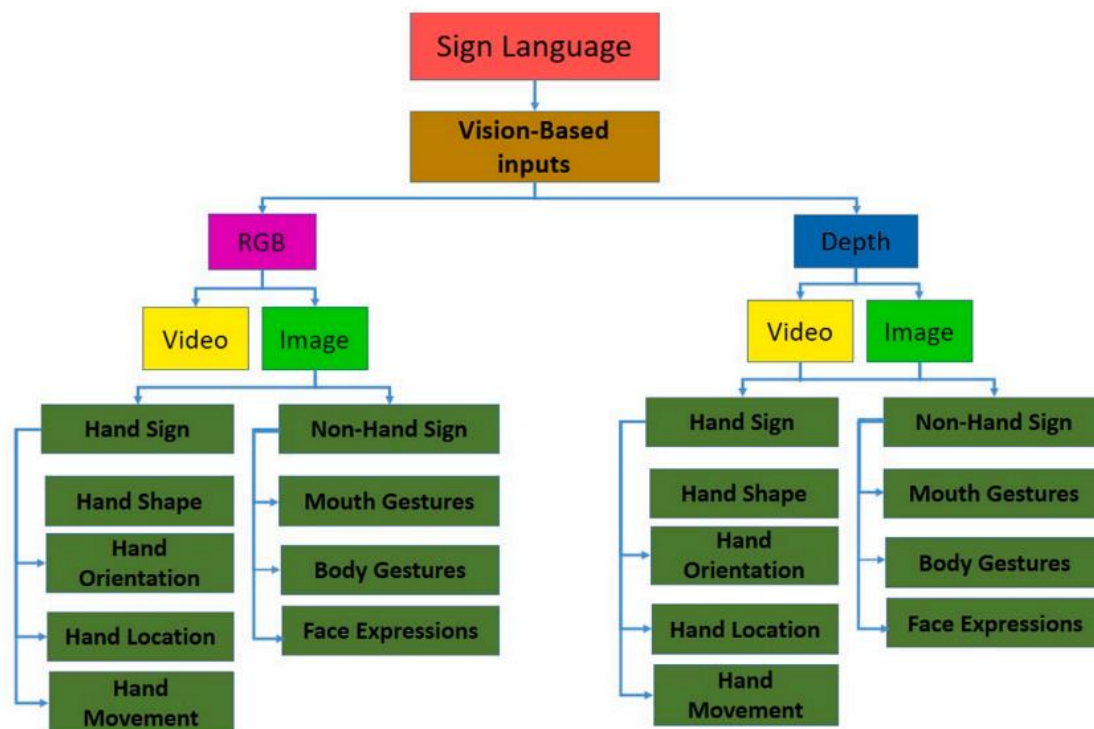


Ilustración 4 Clasificación de modelos de lenguaje de signos basados en visión (Rastgoo, et al., 2021)

La Ilustración 4 deja claro que, para reconocer de forma correcta la totalidad del vocabulario de una lengua de signos a través de medios visuales, se deben tomar en cuenta múltiples partes del cuerpo. En este sentido, “los hablantes de la lengua de señas o señantes coordinan las actividades de las manos, brazos, torso, cara y cabeza para producir grupos de visibles gestos físicos, los cuales son reconocibles como señas” (Cortés & Barreto, 2013).

Si bien el vocabulario de cualquier lengua de signos predomina el uso de las manos y brazos para gesticular palabras y letras, las expresiones faciales muchas veces ayudan a definir el contexto mediante el cual se están expresando las palabras. Dado que la pose de la cara incluye algunas características gramaticales y prosódicas importantes, el uso de estas características como características complementarias con las características de la mano podría mejorar la precisión del reconocimiento del lenguaje de señas (Rastgoo, et al., 2021).

Para el reconocimiento visual de cada uno de estos elementos, se pueden utilizar distintas herramientas. Mohamed, et al. (2021) indican que se utilizan cuatro enfoques diferentes mediante los cuales se pueden utilizar medios visuales para la detección de gestos, siendo estos una cámara de vídeo común, cámaras de rastreo de movimiento, técnicas invasivas y estéreo cámaras.

2.4.1.1 Uso de Cámaras de Vídeo Convencionales para la Detección de Gestos

El uso de cámaras de vídeo para la detección de gestos en lengua de signos es uno de los más comunes actualmente, en la Ilustración 5 se muestra la distribución de herramientas para la captación de gestos para la detección de lengua de signos, según Mohamed, et al. (2021), esta distribución muestra una amplia predominancia de las cámaras en comparación con otras herramientas. Asimismo, Mohamed, et al. (2021) indica que las cámaras digitales o de vídeo

sin dispositivos sensores adicionales son las preferidas por los investigadores para detectar regiones de la mano, ya que son rentables y realistas.

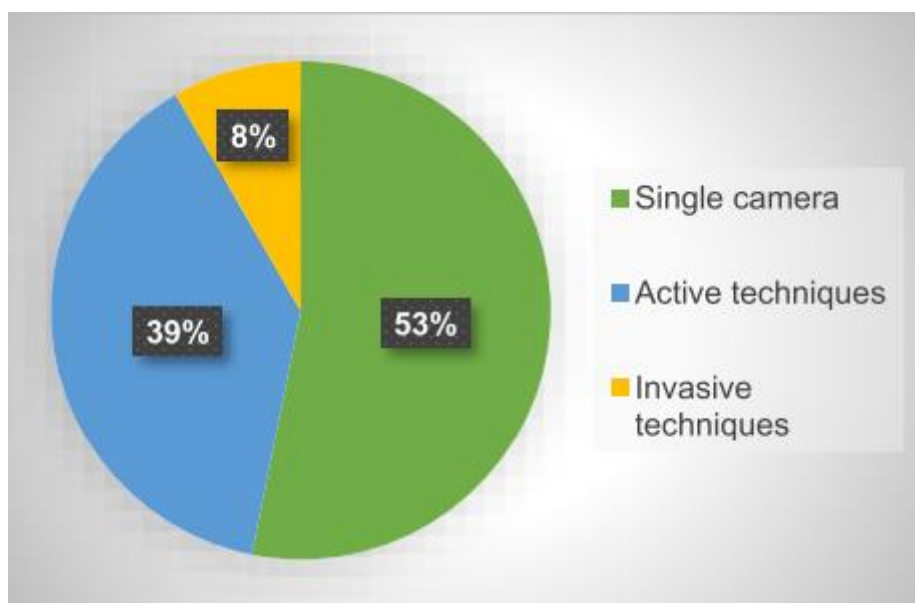


Ilustración 5 Distribución de los estudios relacionados con la detección de gestos para lenguas de signos en el periodo comprendido entre los años 2014 y 2020, tomado de Mohamed, et al. (2021).

Una de las soluciones propuestas para la detección de gestos en lengua de signos es el uso de modelos holísticos para la detección de partes del cuerpo de los signantes para la generación de datos para luego ser procesados e identificados por una red neuronal recurrente. Samaan, et al. (2022) divide su sistema de reconocimiento de gestos en tres segmentos: los datos de entrada, la extracción de características y el modelado.

Los datos de entrada pueden ser tomados de cualquier fuente que tenga videos de personas haciendo gestos en lengua de signos, generalmente son desde una toma frontal en la que se permitan ver los brazos, manos y torso de los signantes; debido a esto, distintos estudios han planteado múltiples formas de generar los datos de entrenamiento para el reconocimiento de signos. Por un lado, Samaan, et al. (2022) generó un conjunto de datos que consiste en 750 videos tomados a 30 fotogramas por segundo. Por otra parte, Kothadiya, et al. (2022) generó un conjunto de datos de 16 hombres y mujeres de entre 20 y los 25 años, las muestras de video contienen una media de 28 fotogramas por segundo, una resolución de 1920×1080 pixeles y con una duración media de 2 segundos.

En cuanto a la extracción de características, el proceso dependerá de las partes del cuerpo que se desean capturar para la detección de gestos, las personas que utilizan lengua de signos utilizan principalmente sus manos para comunicarse, sin embargo, tanto el torso como el rostro ayudan a definir el contexto de la conversación que se está teniendo. Al momento de hablar en lengua de signos “la emisión de signos gestuales no se limita sólo al movimiento de una parte del cuerpo, sino que, [...], intervienen la cabeza, los ojos, las cejas, la boca y las diversas formas que pueden adoptar las manos” (Rodríguez M. Á., 1992).

Para la detección de todas las partes del cuerpo involucradas en la lengua de signos se requiere de un mecanismo que permita distinguir en dos o tres dimensiones cada una de las partes del signante, como ser los brazos, el torso y el rostro. Una de las herramientas más utilizadas actualmente para este propósito es Google MediaPipe, este es conjunto de bibliotecas y herramientas

para que pueda aplicar rápidamente técnicas de inteligencia artificial y aprendizaje automático en cualquier aplicación móvil, web y de escritorio (Google, s.f.).

En el caso de Google MediaPipe, existen tres bibliotecas enfocadas para detectar las partes del cuerpo de las personas al momento de comunicarse en lengua de signos, estas librerías se enfocan en identificar los puntos de importancia de las manos, el rostro y el torso de las personas, visualmente estas librerías se representan en la Ilustración 6.

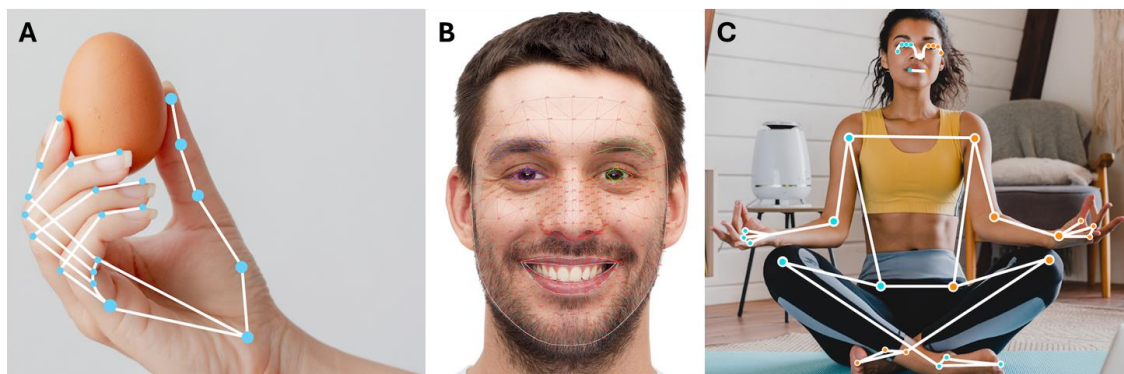


Ilustración 6 Detección de puntos de importancia en distintas partes del cuerpo utilizando Google MediaPipe para (A) las manos, (B) el rostro y (C) el torso/pose (Google, s.f.).

En el estudio realizado por Samaan, et al. (2022), la extracción de características fue mediante Google MediaPipe, utilizando las librerías antes mencionadas se extraen los puntos clave para las tres dimensiones espaciales (X, Y, Z) de ambas manos y el torso para la estimación de cada fotograma de los vídeos generados para el conjunto de datos de la lengua de señas utilizada. Por otra parte, Kothadiya, et al. (2022) utilizó el modelo InceptionResNetV2 para extraer las características principales de las partes del cuerpo de los signantes. Inicialmente, se cambia el tamaño de los fotogramas de vídeo y se transfieren al modelo inceptionResNetV2, el cual fue preentrenado para extraer las características de los gestos, el resultado del modelo inceptionResNetV2 es una matriz que consta del vector de características que representan las partes del cuerpo de los signantes Kothadiya, et al. (2022).

El modelo inceptionResNetV2 es una variante de las redes residuales, siendo estas un tipo de red neuronal artificial utilizada principalmente en el campo de la computación visual y que desde su concepción en 2015 es conocida por abordar el problema del entrenamiento de redes profundas mediante la introducción de un concepto llamado aprendizaje residual. He, et al. (2015) propusieron esta arquitectura haciendo énfasis en que, a diferencia de las redes neuronales convencionales, en las redes neuronales residuales se realizan conexiones de acceso directo entre las capas ocultas que convierten la red en su versión residual homóloga, estos accesos o atajos de identidad se pueden utilizar directamente cuando la entrada y la salida son de las mismas dimensiones.

Finalmente, para la detección de los gestos en lengua de signos, se debe definir un modelo que permita predecir de forma confiable cada uno de los gestos obtenidos a través de la cámara y procesados en la extracción de características. Samaan, et al. (2022) propone el uso de tres redes neuronales recurrentes para este propósito, siendo estas unidades recurrentes cerradas, redes de memoria a corto plazo extendida y redes de memoria a corto plazo extendida bidireccionales.

En el caso Kothadiya, et al. (2022), utilizó las redes de memoria a corto plazo extendida y redes recurrentes cerradas para la clasificación de las características generadas por el modelo inceptionResNetV2, según el estudio realizado, estos métodos se utilizan debido a su capacidad para recordar entradas anteriores, la palabra correspondiente se aprende sobre la base de la activación anterior y la función de entrada actual, y los pesos se ajustan en consecuencia de ambas.

Las unidades recurrentes cerradas (también conocidas como GRU del inglés Gated Recurrent Unit) buscan obtener resultados similares a las redes de memoria a corto plazo extendida con un proceso simplificado. Según Rana, et al. (2016), las GRU cuentan con compuertas lógicas que controlan el flujo de la información, estas pueden ser para determinar cuánta información se puede transmitir en la red (update gates o compuertas de actualización) o cuánta información puede omitirse en la red (reset gates o compuertas de olvido).

Las redes de memoria a corto plazo extendida para los estudios antes mencionados fueron utilizadas de forma unidireccional y bidireccional. La diferencia entre una y otra es que la variante bidireccional es una extensión de los modelos de memoria a corto plazo extendida en la que se aplican dos modelos de este tipo a los datos de entrada. En el primer ciclo de ejecución, se aplica un modelo de memoria a corto plazo en la capa de entrada; mientras que en la segunda ejecución se aplica de forma inversa la secuencia de entrada que se introduce en el modelo de red neuronal (Siarni-Namini, et al., 2019).

En el caso Kothadiya, et al. (2022), se utilizó una red de memoria a corto plazo extendida unidireccional para procesar la información obtenida de la extracción de características mediante inceptionResNetV2, para posteriormente trasladar el resultado obtenido de la red neuronal a corto plazo hacia un modelo de unidad recurrente cerrado, como se puede apreciar en la Ilustración 7.

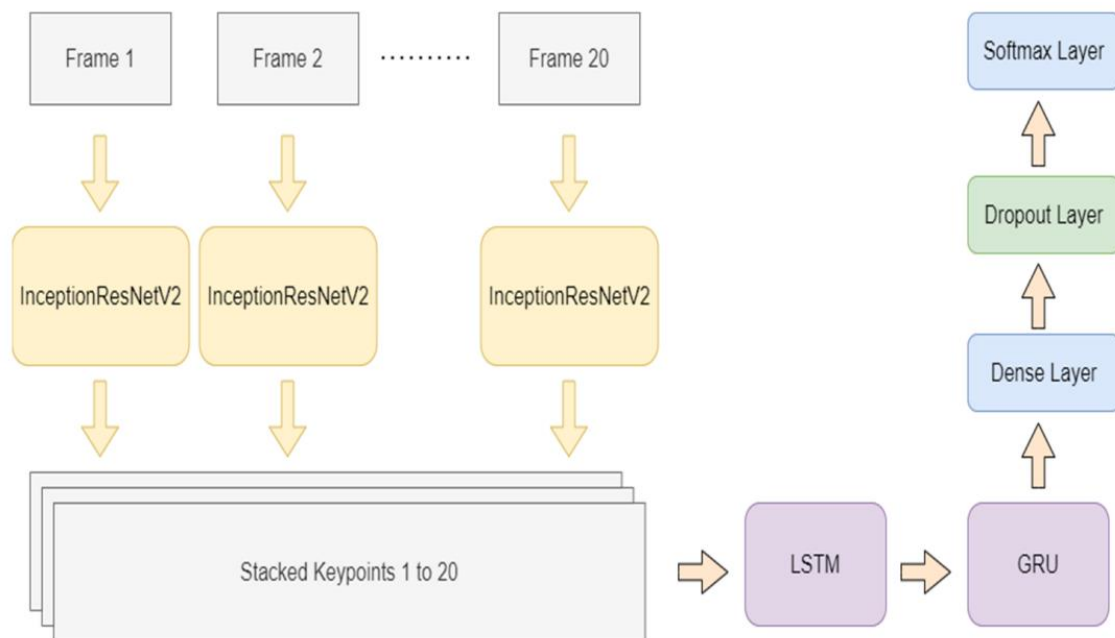


Ilustración 7 Arquitectura utilizada para la detección de gestos en lengua de signos mediante los modelos InceptionResNetV2, redes de memoria a corto plazo extendida y unidades recurrentes cerradas, tomado de Kothadiya, et al. (2022).

Finalmente, al momento de comprar el rendimiento de los tres modelos hasta ahora mencionados, estos se ven influenciados por las características de la

lengua de signos que se busca interpretar. En el estudio realizado por Samaan, et al. (2022) se realizó una detección de gestos en lengua de signos considerando el torso y los brazos y una detección de gestos considerando torso, brazos y rostro. Después de realizar los dos experimentos, se encontró que la precisión es muy cercana en ambos; por lo tanto, la técnica preferida depende de la situación problemática. Por ejemplo, si el modelo se va a utilizar en una lengua de signos que depende de la expresión facial, es preferible utilizar puntos faciales (Samaan, et al., 2022). Esta similitud en rendimiento se evidencia en la Tabla 4.

Modelo	GRU	LSTM	BILSTM
Experimento 1: Datos con datos del torso y brazos			
Precisión de entrenamiento	100%	99.9%	99.9%
Precisión de verificación	100%	99.6%	99.3%
Número de épocas de entrenamiento	241	65	75
Experimento 2: Datos con datos del torso, brazos y rostro			
Precisión de entrenamiento	100%	99%	99.9%
Precisión de verificación	100%	99.6%	99%
Número de épocas de entrenamiento	250	36	49

Tabla 4 Cuadro comparativo de la precisión en el entrenamiento y verificación de los modelos de unidad recurrente cerrada (GRU), red de memoria a corto plazo extendida unidireccional (LSTM) y red de memoria a corto plazo extendida bidireccional (BILSTM) en los experimentos ejecutados tomando en consideración brazos, torso y rostro, basado en Samaan, et al. (2022).

Un aspecto importante para mostrar en la Tabla 4, es que, aunque el rendimiento de la red neuronal de memoria a corto plazo extendida unidireccional es ligeramente inferior a los otros modelos evaluados, este requiere de una menor cantidad de épocas de entrenamiento, por lo que requiere de menos entrenamiento para llegar a una precisión aceptable.

En términos generales, una época de entrenamiento hace referencia a un ciclo completo por todo el conjunto de datos de entrenamiento en una red neuronal. En el contexto del aprendizaje automatizado y la inteligencia artificial es un concepto crucial en el proceso de entrenamiento de un modelo. Una época de entrenamiento significa que cada muestra del conjunto de datos destinado al entrenamiento del modelo ha tenido la oportunidad de actualizar los parámetros internos del modelo (pesos), de forma que el algoritmo de aprendizaje se ejecute hasta que el error del modelo se haya minimizado lo suficiente (Brownlee, 2018).

Para concluir, los sistemas de detección de gestos en lengua de signos mediante técnicas de computación visual representan un medio práctico para la interpretación de lengua de signos. La gran mayoría de dispositivos hoy en día cuentan con una cámara de vídeo que, en combinación con modelos holísticos y técnicas de aprendizaje automatizado, pueden ser utilizadas para la interpretación de gestos en lengua de signos sin necesidad de contar con otro equipo especializado, como dispositivos similares al Microsoft Kinect o guantes especializados con sensores.

2.4.1.2 Uso de Dispositivos de Rastreo de Movimientos para la Detección de Gestos

Existen dispositivos que facilitan el seguimiento del cuerpo mediante cámaras y sensores, uno de los más populares es Microsoft Kinect. Este es un equipo

desarrollado por la empresa Microsoft con sensores que permiten hacer rastreo de distintas partes del cuerpo, como la cabeza, torso, brazos y piernas.

A pesar de que Microsoft discontinuó la fabricación de dispositivos Microsoft Kinect, las unidades existentes siguen siendo de utilidad para múltiples estudios, entre ellos el reconocimiento de gestos. Según Mohamed, et al. (2021), el uso de Microsoft Kinect para reconocer los gestos de las manos es para hacer que el sistema sea más confiable, este dispositivo tiene la capacidad de capturar información de profundidad, haciendo que entidades extraídas de los sensores son resistentes a los cambios de iluminación, tamaño y rotación.

Uno de los ejemplos existentes es el propuesto por Raghuveera, et. al (2020), en el cual se propone un sistema de reconocimiento de lengua de signos india. El sistema propuesto consta de cinco etapas: preprocesamiento, segmentación de las manos, extracción de características y reconocimiento de signos, predicción del signo de salida mediante la técnica de ensamble e interpretación de oraciones Raghuveera, et. al (2020).

El preprocesamiento de los datos se divide en dos pasos: adquisición de imágenes y la eliminación de ruido en las imágenes. Para la adquisición de imágenes, se seleccionaron 140 signos estáticos de la lengua de signos india, realizados por 21 voluntarios con diferentes variaciones posibles, lo que representa hasta 40 imágenes de cada signo Raghuveera, et. al (2020). Una vez obtenidas estas imágenes, se eliminaba el ruido y posibles distorsiones de cada una de estas, a modo de tener una imagen de profundidad con una resolución de 640x480 píxeles. El resultado de este preprocesamiento se aprecia en la Ilustración 8.

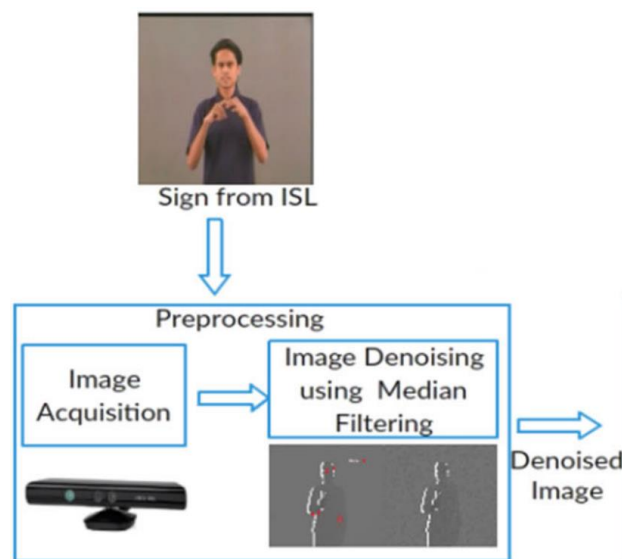


Ilustración 8 Preprocesamiento de imágenes a través de un dispositivo Microsoft Kinect, tomado de Raghuveera, et. al (2020).

Para la segmentación de las manos en el sistema propuesto por Raghuveera, et. al, se separaron las regiones de interés en la imagen de entrada, misma que fue preprocesada en el paso anterior, específicamente en las manos y codos. Para lograr esta segmentación en tiempo real, la segmentación debe ser rápida; por lo que se desarrolló el agrupamiento (clustering) de K-mean para segmentar la imagen en clústeres más cercanos Raghuveera, et. al (2020). Una vez definidos los clústeres, se realiza el cálculo de máximos locales e inicialización de la cantidad K de clústeres con máximos locales. Este proceso se resume en la Ilustración 9.

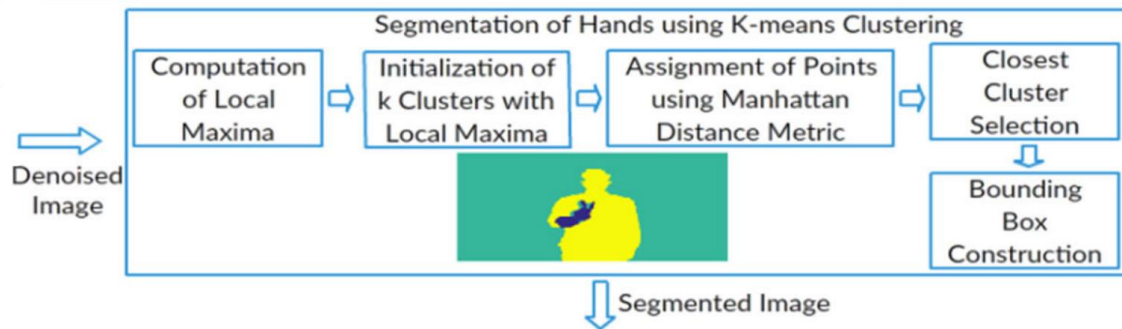


Ilustración 9 Proceso de segmentación a través de clusterización K-means según Raghuveera, et. al (2020).

Seguidamente, la imagen segmentada pasa por un procesamiento de extracción de características, la cual consiste en implementar distintas técnicas para la generación de una serie de vectores que facilitarán la identificación y reconocimiento de los gestos. Raghuveera, et. al (2020) utiliza los patrones locales binarios, histogramas de gradientes orientados y características robustas aceleradas para la extracción de características, como se muestra en la Ilustración 10.

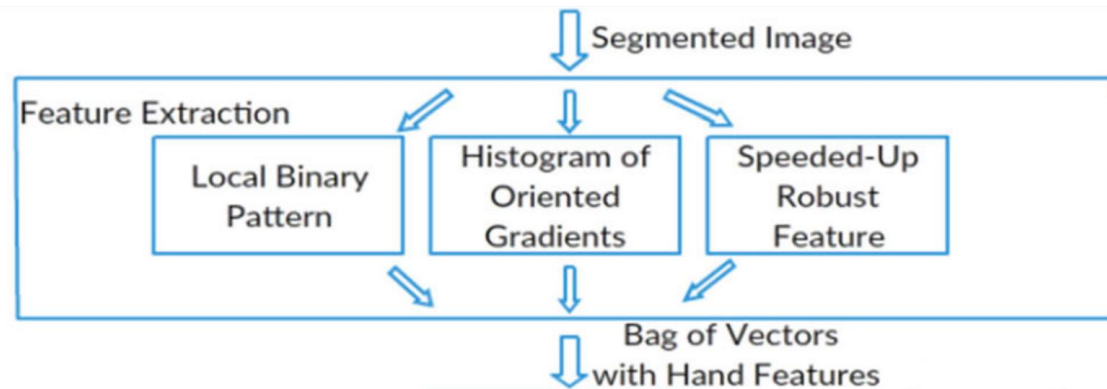


Ilustración 10 Procesamiento de extracción de características para la generación de vectores de características de las manos para el reconocimiento de lengua de signos utilizado por Raghuveera, et. al (2020).

Los patrones binarios locales son utilizados para la clasificación de texturas en el procesamiento de imágenes y la visión artificial, siendo especialmente útil para resumir las estructuras locales en una imagen comparando cada píxel con su vecindario circundante. Los patrones binarios locales funcionan con mucho éxito en diversos dominios, como la clasificación de texturas, la detección de cuerpos humanos y el reconocimiento facial Chen, et al. (2013).

Los histogramas de gradientes orientados son un descriptor de características utilizado en la visión por computador y el procesamiento de imágenes para la detección de objetos. Es especialmente adecuado para tareas como la detección humana en imágenes y videos debido a su eficacia en la captura de información de la estructura de los bordes y el gradiente. De acuerdo con Zhang et al. (2020), en los histogramas de gradientes normalizados, las entidades se representan mediante un histograma de la información de cambio de dirección de los bordes en la imagen, primero se calculan los gradientes, seguidamente se clasifican los bordes por orientación para finalmente ser normalizados y clasificados. Este proceso se describe de forma visual en la Ilustración 11.

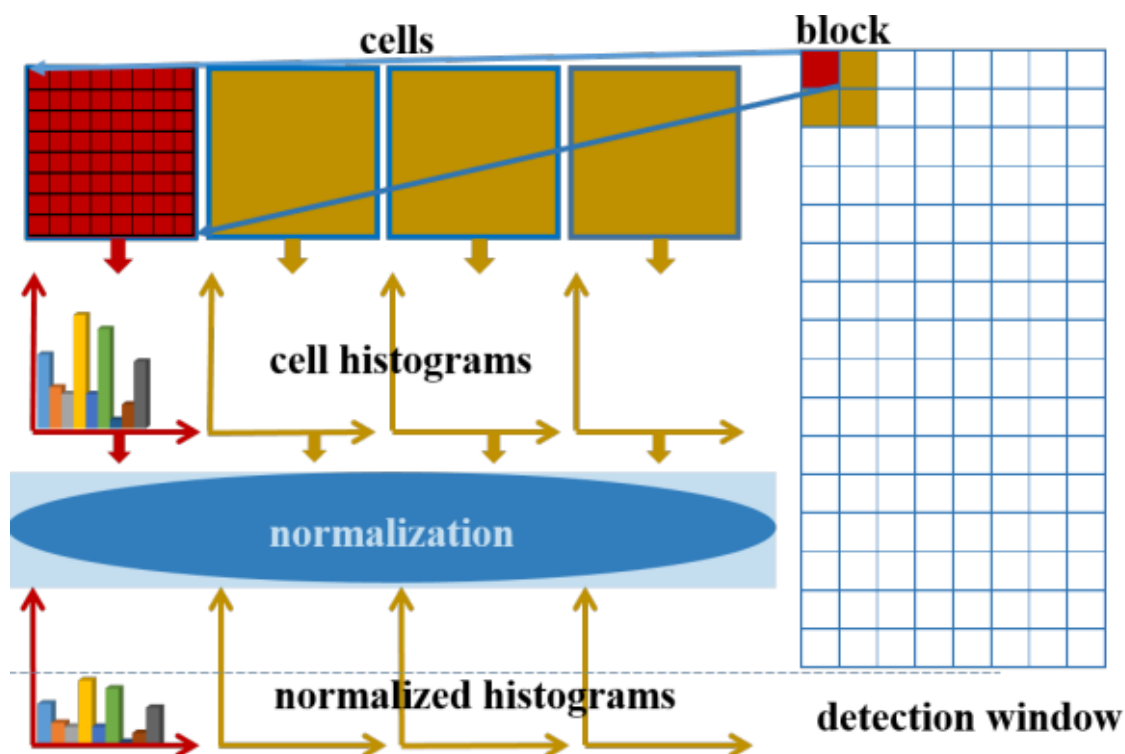


Ilustración 11 El proceso de la canalización de extracción de características mediante histogramas de gradientes orientados, tomado de Zhang et al. (2020).

Las características robustas aceleradas son un método detector y descriptor de características locales, que se utiliza principalmente en tareas de visión por computador, como el reconocimiento de objetos, el registro de imágenes y la reconstrucción 3D. De acuerdo con Bansal et al. (2021), las características robustas aceleradas producen mejores resultados en rotación, desenfoque e iluminación, tiene un mejor rendimiento en cuanto a la velocidad del algoritmo y la longitud del vector de características es más corta que las características extraídas de otras técnicas como la transformación de entidad invariable de escala.

Seguidamente a la extracción de características, Raghuveera, et. al (2020) utilizan un clasificador de máquina de vectores de soporte, el cual utiliza los vectores generados por los métodos descritos anteriormente para la clasificación de etiquetas para cada uno de los gestos obtenidos en las imágenes del Microsoft Kinect, en la Ilustración 12 se muestra el diagrama mediante el cual se procesa la clasificación de las características obtenidas. El potencial de las máquinas de vectores de soporte se deriva de su capacidad para aprender patrones de clasificación de datos manteniendo un balance entre precisión y reproducibilidad, este algoritmo se ha convertido en una herramienta ampliamente utilizada para la clasificación en múltiples escenarios de ciencia de datos (Pisner & Schnyer, 2020).

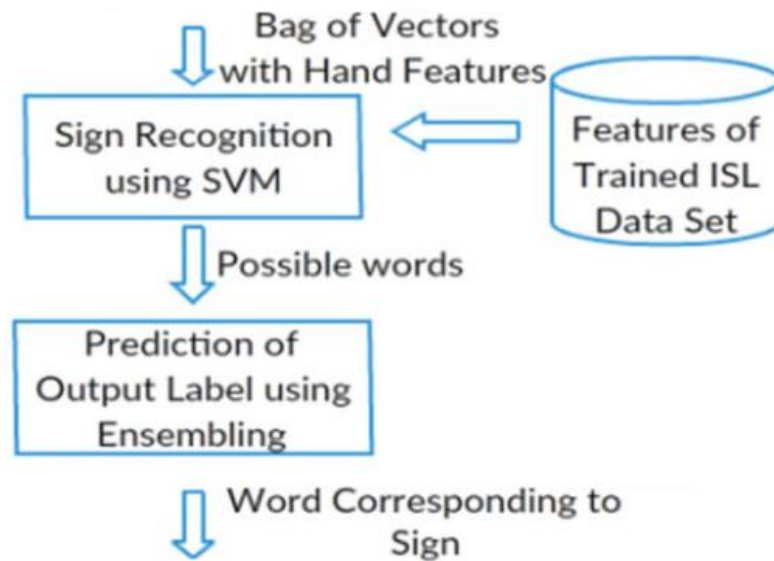


Ilustración 12 Clasificación de los vectores obtenidos de la extracción de características utilizando máquinas de vectores de soporte, tomado de Raghuveera, et. al (2020).

Para lograr una clasificación efectiva con las máquinas de vectores de soporte, se debe llevar a cabo un proceso de entrenamiento previo con datos debidamente etiquetados, ya que esta técnica forma parte de la categoría de aprendizaje supervisado. En el ejemplo de Raghuveera, et. al (2020), el conjunto de datos de la lengua de signos india consta de ciento cuarenta signos estáticos seleccionados a mano, realizados por veintiún voluntarios con diferentes variaciones posibles, lo que representa hasta 40 imágenes debajo de cada signo, llegando a un total de cuatro mil seiscientas imágenes; de esta cantidad, el ochenta por ciento de los datos fueron utilizados para el entrenamiento y el veinte por ciento de los datos se utilizaron para la validación del clasificador.

Finalmente, una vez se tienen identificadas las palabras expresadas a través de los gestos, se ejecuta un proceso de interpretación de las palabras obtenidas para la generación de una frase en la lengua de signos india. Raghuveera, et. al (2020) realiza esta interpretación utilizando índices invertidos, según ellos, esta técnica es la más apropiada para dar una solución a la interpretación de oraciones basándose en el vocabulario limitado que tienen en su base de datos, este proceso se describe en la Ilustración 13.

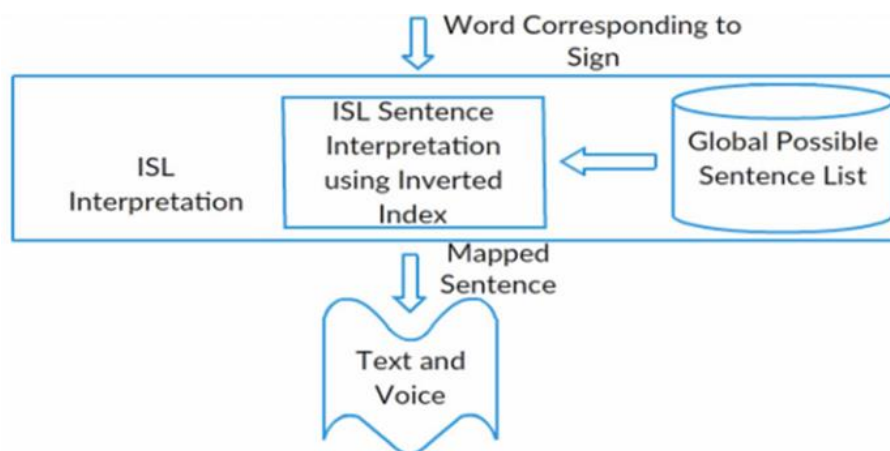


Ilustración 13 Proceso de interpretación de palabras detectadas en lengua de signos india, tomado de Raghuveera, et. al (2020).

La técnica de índices invertidos se basa en una estructura de datos que se utiliza en los sistemas de recuperación de información para almacenar de manera eficiente una asignación de contenido, en el contexto del trabajo planteado por Raghuveera, et. al (2020), se almacenan las palabras identificadas anteriormente y se comparan con una lista de posibles frases relacionadas a las palabras identificadas.

La interpretación de las frases, según Raghuveera, et. al (2020), consiste en dos pasos, el primero es crear la base de datos con las frases que utilizarán los índices invertidos para comparar cada palabra detectada, mientras que el segundo y último paso es comparar estas palabras con la base de datos generada, en la Tabla 5 se muestra la lista de palabras clave con su respectiva interpretación en lengua de signos india.

No.	Palabras clave identificadas	Posible interpretación en lengua de signos india
1	Mail, me	Send me a mail
2	I, fine	I am fine
3	Sample, leprosy	The sample shows signs of leprosy
4	How, you	How are you
5	B, bottle	B for Bottle
6	Blow, leaf	Blow the leaf away
7	You, strong	You are strong
8	What, weigh	What is your weight?
9	Vacation, fine	The vacation was fine
10	Unit, strong	Unity is strength
11	Boxing, strong	Boxing makes us strong
12	Keep, up	Keep it up
13	Bud, perfect	The bud looks perfect
14	Appreciation, work	Appreciation of work
15	Deposit, thick, density	The deposit has thick density
16	I, mail, you	I will mail you
17	Measure, density	Measure the density
18	Negative, opposite, good	Negative is the opposite of good
19	Snake, bite, me	A snake bit me
20	I, lost, weight	I have lost my weight
21	Add, small, quantity, sugar	Add small quantity of sugar
22	Learn, respect	Learn to respect
23	I, good, friend	I am a good friend
24	Buds, specific, season	These buds are specific to a season
25	Opposition, control	Opposition is in control

No.	Palabras clave identificadas	Posible interpretación en lengua de signos india
26	Pray, Varanasi	Pray at Varanasi

Tabla 5 Lista de frases para la interpretación de gestos en lengua de signos india a frases en inglés, tomado de Raghuveera, et. al (2020).

Otra técnica utilizada para el reconocimiento de gestos en la lengua de signos es mediante el uso de controladores de movimiento (Leap Motion Controllers). Según Mohamed, et al. (2021), el uso de controladores de movimiento puede apuntar explícitamente al seguimiento de los movimientos de las manos y los dedos en formato digital en tres dimensiones, asimismo, puede facilitar la identificación de las articulaciones de los dedos, así como rastrear los movimientos del signante mientras se realizan gestos.

Si bien el uso de controladores de movimiento puede por sí solos ayudar a detectar gestos en lengua de signos, cuando se combina con tecnologías como la de Microsoft Kinect se pueden obtener mejores resultados. Según Kumar, et al. (2017) combinando los datos adquiridos con ambos sensores (Microsoft Kinect y un controlador de movimiento), se consigue una mayor precisión en los resultados; esto gracias a la ayuda de combinar los datos generados por ambos sensores con una combinación de clasificadores, esta diferencia se muestra en la Tabla 6.

No.	Clasificador utilizado	Controlador de movimiento	Microsoft Kinect	Controlador de movimiento + Microsoft Kinect
Gestos con una mano				
a	Modelo oculto de Markov	95.12%	82.86	97.38
b	Red neuronal de memoria a corto plazo extendida	79.41%	84.74	87.62
Gestos con dos manos				
c	Modelo oculto de Markov	92.73%	79.85	93.64
d	Red neuronal de memoria a corto plazo extendida	81.82%	72.3%	83.49%
Gestos combinados				
e	Modelo oculto de Markov	92.6%	84.26%	95.6%
f	Red neuronal de memoria a corto plazo extendida	72.39%	78.66%	84.57%

Tabla 6 Porcentaje de precisión en el reconocimiento de gestos evaluando la comunicación con una mano y dos manos después de la combinación de funciones, tomado de Kumar, et al. (2017).

Haciendo una comparativa entre las técnicas aplicadas en ambos modelos de detección de gestos mediante el uso de dispositivo de rastreo de movimientos, estos ofrecen una solución de detección indirecta de los gestos en lengua de signos. El rendimiento de los modelos utilizados en los estudios descritos representa se encuentran dentro de un rango aceptable para una interpretación confiable de los gestos, especialmente cuando se combinan las facilidades del

Microsoft Kinect con los controladores de movimiento, tal como lo propone Kumar, et al.

2.4.1.3 Uso de técnicas invasivas para la detección de gestos

Una técnica más directa para la detección de gestos en lengua de signos es el uso de dispositivos que detecten el movimiento de las manos para la interpretación de los gestos realizados por los signantes. Varios artículos utilizan técnicas invasivas como guantes para extraer imágenes de gestos con las manos, aunque el uso de guantes de color ayuda al proceso de captura de imágenes hasta cierto punto, otros estudios combinan sensores como Microsoft Kinect con un par de guantes de color para extraer características como ser la profundidad, el movimiento y el color (Mohamed, et al., 2021).

El uso de guantes para la detección de gestos en lengua de signos es una práctica común en esta área de estudio. A pesar de que, según el estudio Mohamed et al. (2021), esta técnica fue la menos utilizada en comparación con otras técnicas como métodos visuales de detección, la cantidad total de estudios realizados entre los años 2014 y 2020 suma un total 8 artículos, como máximo; otros estudios indican que en los últimos años se ha incrementado el número de estudios utilizando dispositivos como guantes para la detección de gestos en lengua de signos.

En un estudio realizado por Saeed, et al. (2022), en el que se exploraron un total de 1,120 artículos en diversos portales científicos entre los años 2017 y 2022, un total de 115 estudios fueron detectados para el análisis cualitativo objeto de dicho estudio, todos referentes al uso de guantes para la detección de gestos en lengua de signos. De acuerdo la Ilustración 14, la mayoría de los estudios se enfocaron en los gestos utilizando una sola mano, siendo un 76 por ciento de los artículos enfocados en estas técnicas, mientras que el 24 por ciento restante se enfocó en gestos con dos manos. Lo anterior representa una limitación en la implementación real de este tipo de tecnologías para la interpretación en tiempo real de la lengua de signos.

Número de manos tomados en cuenta
para la detección de gestos

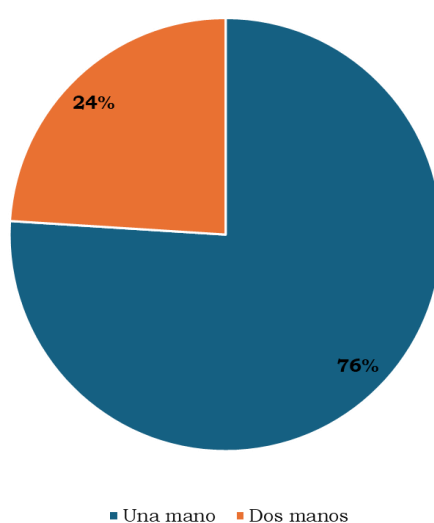


Ilustración 14 Porcentaje de estudios realizados para la detección de lengua de signos tomando en consideración con una o dos manos, basado en Saeed, et al. (2022).

Otro aspecto para considerar en este tipo de implementaciones es el alcance que tienen los dispositivos de detección de gestos en lengua de signos. De acuerdo con la Ilustración 15, en la cual se ilustra el número de artículos y tipos de gestos cubiertos por trabajos y artículos objeto del estudio elaborado por Saeed, et al. (2022), la mayoría de los estudios en detección de gestos para la lengua de signos utilizando guantes para detectar estos gestos se enfocan en palabras concretas y letras del alfabeto, mientras que una menor parte se enfoca en la elaboración de estructuras gramaticales más complejas, como las oraciones.

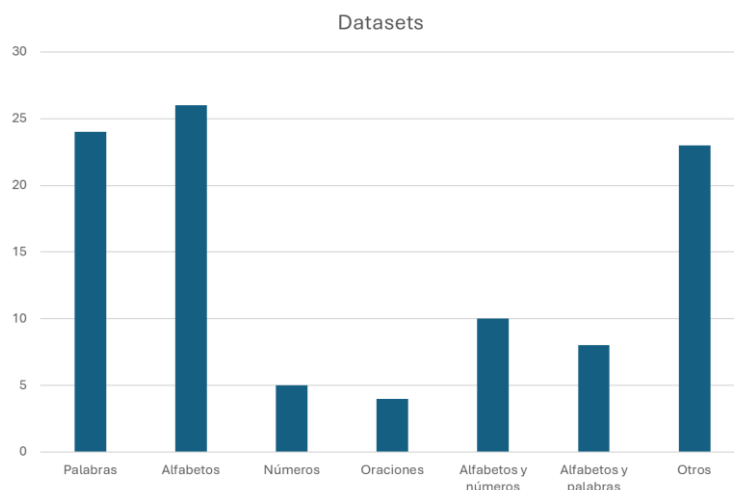


Ilustración 15 Tipos de gestos en los cuales se enfocan los artículos científicos estudiando la detección de gestos en lengua de signos mediante guantes con sensores de presión y movimiento, basado en Saeed, et al. (2022).

Un ejemplo de esta implementación para la detección de gestos es la propuesta por Jeyasheeli & Indumathi (2021) para la detección de lengua de signos india, el cual consiste en dos elementos que se encargan de identificar los movimientos de los signantes. En primer lugar se cuenta con un guante que detecta los movimientos de las manos, mientras que la detección de los gestos se complementa con un módulo de detección de los movimientos para ser interpretados en lengua de signos.

En el caso de los guantes, estos están equipados con sensores de movimiento que detectan las flexiones de los dedos, también detectan presión y movimiento de las manos mediante un giroscopio. En la Ilustración 16 se muestra la instalación del dispositivo propuesto, sobre las palmas de ambas manos, se instala el sensor de presión, luego se conecta una resistencia al sensor de presión, para garantizar que el flujo de corriente se estabilice, uno de los sensores se monta en una resistencia, finalmente los sensores se interconectan al microcontrolador Arduino con el fin de obtener mediciones precisas para cada signo generado por las personas utilizando los guantes (Jeyasheeli & Indumathi, 2021).



Ilustración 16 Configuración de hardware instalado en los guantes para la detección de lengua de signos india, tomado de Jeyasheeli & Indumathi (2021).

La información generada por estos sensores es recolectada para la creación del conjunto de datos que se utilizará como conjunto de entrenamiento para la red neuronal utilizada para la interpretación de los gestos en lengua de signos. La información capturada del sensor se utiliza en el módulo de identificación de gestos para clasificar e identificar el gesto, para la categorización de los gestos generados por los signantes se analiza la información del sensor mediante una red neuronal convolucional (Jeyasheeli & Indumathi, 2021).

El conjunto de datos utilizado para el estudio de Jeyasheeli & Indumathi (2021) constó de 50 gestos generados por 50 signantes diferentes, cada gesto se ejecutó en una ventana de 10 segundos por signo. Para el proceso de entrenamiento se siguió una práctica habitual en el aprendizaje automatizado, la cual consiste en inicializar la red neuronal convolucional con pesos generados aleatoriamente, y a medida se ejecuta el entrenamiento de la red estos pesos se van ajustando a medida se calcula el error entre los valores esperados y los valores generados en ese ciclo de entrenamiento.

Para esta implementación se contó con un conjunto de datos de 2.500 elementos que representan 50 signos diferentes de la lengua de signos india. Para una mayor precisión, se conserva una determinada proporción del conjunto de datos. En este proyecto, la proporción de entrenamiento y prueba se definió primero en 60:40 (60 por ciento para el conjunto de entrenamiento y 40 por ciento para el conjunto de verificación) y luego en 80:20 (80 por ciento para el conjunto de entrenamiento y 20 por ciento para el conjunto de verificación) para lograr una mayor precisión (Jeyasheeli & Indumathi, 2021). Como se ilustra en la Tabla 7, la mayor precisión se obtuvo con la proporción 80:20, con un valor de 97,8 por ciento.

Proporción del conjunto de datos	Caso 1 Proporción 60:40	Caso 2 Proporción 80:20
Número total de muestras en el conjunto de datos	2.5	2.5
Conjunto de entrenamiento	1.5	2
Conjunto de verificación	1	500
Precisión de la red neuronal convolucional	89,0%	97,8%

Tabla 7 Precisión en ambas proporciones definidas para la red neuronal convolucional utilizada para la detección de gestos en lengua de signos india mediante la implementación de guantes con sensores de movimiento, basado en Jeyasheeli & Indumathi (2021)

Además de los sensores de presión utilizados en el estudio mencionado anteriormente, otras de las tecnologías utilizadas para los guantes de detección de gestos son los sensores de flexión. Los sensores de flexión son uno dispositivos que miden la cantidad de flexión que experimenta el cuerpo en el que se encuentran, en este caso específico, el nivel de flexión que experimentan los dedos de las manos. Una aplicación de esta tecnología en la detección de gestos es la propuesta por Yin, et al. (2018), el cual adopta un algoritmo para la detección de gestos utilizando una técnica de coincidencia de plantillas (template matching) y redes neuronales.

A diferencia del sistema de detección de gestos de Jeyasheeli & Indumathi, el sistema desarrollado por Yin, et al. reduce la complejidad del dispositivo al enfocarse únicamente en los sensores de flexión (un sensor por cada dedo de la mano), los sensores son una parte importante de los guantes de datos, ya que afectan a la precisión de los datos recopilados, por lo que afectan a la precisión del reconocimiento de gestos (Yin, et al., 2018), esta implementación se puede muestra en la Ilustración 17.

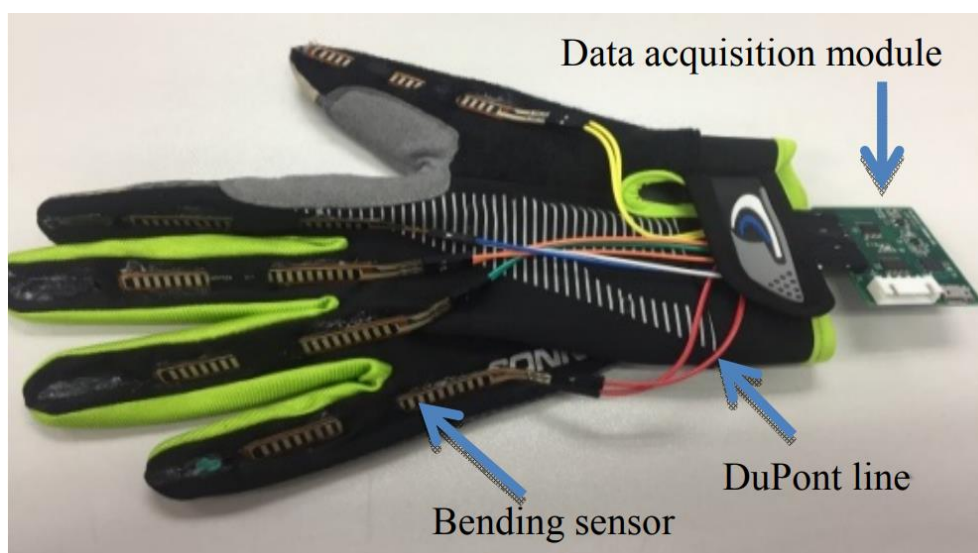


Ilustración 17 Implementación de guante para la detección de gestos, el cual consiste en sensores de flexión conectados a un módulo de adquisición de datos mediante un cable conectado a los sensores, tomado de Yin, et al. (2018).

De igual forma, el modelo utilizado para la detección de gestos es distinto al propuesto por Jeyasheeli & Indumathi. En este caso, Yin, et al. utilizaron dos algoritmos para la detección de gestos, el primero se denomina un modelo de coincidencia de plantillas, mientras que el segundo modelo es una red neuronal de propagación hacia atrás. De acuerdo con Yin, et al. (2018), los experimentos muestran que la combinación de ambos algoritmos o modelos puede mejorar la precisión del reconocimiento y acortar el tiempo de entrenamiento para la detección de gestos.

La coincidencia de plantillas, también conocido como “template matching”, es un método de reconocimiento de patrones genérico que originalmente fue utilizado para el procesamiento de imágenes, la plantilla es conocida como “imágenes pequeñas” para buscar. Por lo anterior, Wang, et al. (2016) define al algoritmo de coincidencia de plantillas como el proceso de buscar un objetivo en una imagen grande de acuerdo con el tamaño y la dirección de la plantilla para encontrar el mismo elemento de imagen.

Por otra parte, las redes neuronales con propagación hacia atrás es un tipo de red neuronal artificial que utiliza el algoritmo de propagación inversa para el entrenamiento, este consiste en el cálculo del error entre el resultado obtenido por la red neuronal y el valor esperado por el conjunto de entrenamiento y trasladarlo a los pesos de la red neuronal desde la capa de salida hasta la capa de entrada.

De acuerdo con Yin, et al. (2018), la precisión promedia del algoritmo conjunto (combinación de la coincidencia de plantillas con la red neuronal con propagación hacia atrás) para diferentes detectores es significativamente más alta que la de los dos algoritmos por separado, esta diferencia se aprecia de mejor forma en la Tabla 8.

Origen de datos	Precisión del algoritmo de coincidencia de plantillas	Precisión del algoritmo de redes neuronales con propagación hacia atrás	Precisión de la combinación de ambos algoritmos
Todos los datos	96,7%	98,4%	99,8%
Detector 1	99,6%	100,0%	100,0%
Detector 2	89,3%	99,0%	96,0%
Detector 3	95,9%	100,0%	100,0%
Detector 4	98,8%	99,5%	100,0%
Detector 5	100,0%	100,0%	100,0%

Tabla 8 Comparativa en la precisión de los sensores utilizando los modelos de coincidencia de plantillas, redes neuronales con propagación hacia atrás y la combinación de ambos modelos, cada detector representa un dedo de la mano del signante, basado en Yin, et al. (2018).

Para concluir, los sistemas mostrados que utilizan guantes para la detección de movimientos presentan otro enfoque para afrontar las necesidades en interpretación de lengua de signos. Por un lado, la propuesta de Jeyasheeli & Indumathi presenta una mayor robustez en el funcionamiento del sistema, ya que detecta cambios en el movimiento, flexión y presión de ambas manos del signante, mientras que el sistema Yin, et al. es una versión simplificada que busca detectar gestos en una sola mano mediante sensores de flexión únicamente. En cuanto al rendimiento de los modelos, ambas soluciones ofrecen precisiones aceptables para la detección de gestos en lengua de signos, la mayor diferencia entre ambos está en la clase de modelos que se utilizan para la predicción de los gestos, siendo el modelo de Jeyasheeli & Indumathi el de mayor alcance.

2.4.1.4 Uso de cámaras estereoscópicas para la detección de gestos

Una estereo cámara es un dispositivo que utiliza dos o más lentes para capturar imágenes desde ángulos ligeramente diferentes, imitando la visión binocular del ojo humano. Esta configuración permite que las cámaras capturen información referente a la profundidad de un objeto, lo que permite la creación de imágenes o modelos en tres dimensiones de objetos y escenas.

Las cámaras estereoscópicas son una herramienta adicional para la detección de gestos en lengua de signos. La implementación de un sistema de detección de gestos con las manos mediante cámaras estereoscópicas es útil para determinar la posición de la mano en un espacio en tres dimensiones y la

posición de los dedos mediante la determinación de las puntas de estos (Miron, et al., 2017).

Si bien existen cámaras estereoscópicas que se utilizan en dispositivos de realidad virtual y realidad aumentada, el efecto de estas cámaras puede ser conseguido mediante el uso de múltiples cámaras convencionales. Un ejemplo de esta tecnología aplicada al reconocimiento de gestos en lengua de signos es SUGO, el cual es un sistema basado en vídeo de profundidad para traducir el lenguaje de señas coreano a texto usando la cámara frontal de un teléfono inteligente (Park, et. al, 2021).

De acuerdo con Park, et al. (2021), las cámaras de profundidad de los teléfonos inteligentes tienen varias limitaciones en cuanto a los sensores de profundidad basados en infrarrojos (como el Apple TrueDepth de los iPhones), ya que pueden funcionar mal bajo la luz solar muy brillante, ya que el espectro solar incluye longitudes de onda infrarrojas que interfieren en la detección del cuerpo de los signantes. Debido a este tipo de limitaciones, se requiere de una serie de preprocesamiento de las imágenes obtenidas a través de las cámaras de los dispositivos móviles.

En el caso específico de SUGO, la información obtenida por los sensores del dispositivo móvil pasa por un proceso de preparación previo a la detección de los gestos. Según Park, et al. (2021), el proceso de preprocesamiento tiene como objetivo eliminar el ruido de fondo y enfatizar los gestos humanos para que el contenido central de las imágenes pueda ser explotado de manera efectiva para la traducción del lenguaje de señas. Para llevar a cabo esta función, se elimina el fondo de la imagen obtenida y se pasa por un proceso de renormalización de la imagen, este proceso se muestra en la Ilustración 18.

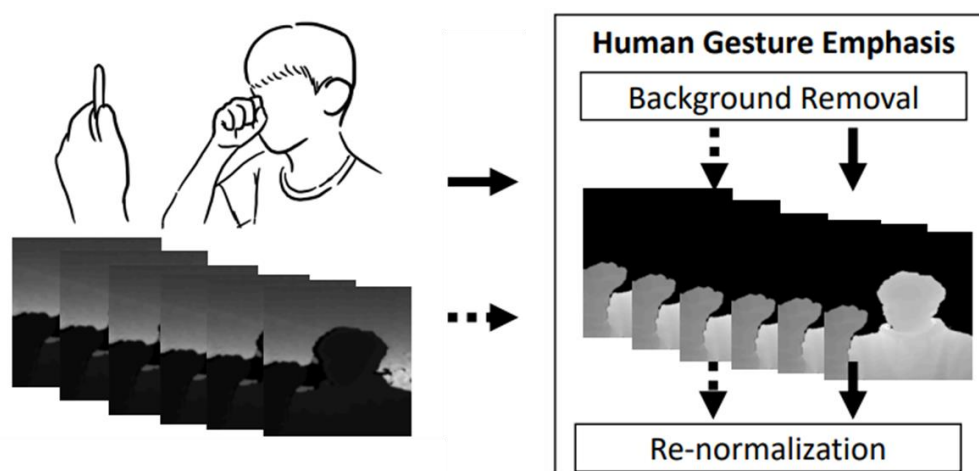


Ilustración 18 Funcionamiento del módulo de énfasis de gestos humanos, el cual se encarga de eliminar el fondo y renormalizar la imagen obtenida por los sensores del dispositivo móvil, tomado de Park, et al. (2021).

Una vez realizado el preprocesamiento de las imágenes obtenidas por los sensores de los dispositivos móviles, estos se almacenan para formar parte del conjunto de datos que sirve como base del entrenamiento del modelo definido para SUGO. Para entrenar y probar el modelo de detección de gestos, se creó un conjunto de datos de vídeo de profundidad de 50 palabras diferentes del vocabulario de lenguaje de signos disponible en el diccionario de lengua de signos coreano (Park, et al., 2021). Las palabras utilizadas se describen en la Tabla 9.

Posiciones de gestos con las manos	Palabras	
Cerca del rostro	•Gustar •Comer •Derecha •Ninguno •Mental •Poder	•Sonreír •Edad •Alto •Aburrido •Duda •Pobreza
Cerca del cuerpo	•1 •2 •3 •4 •5 •6 •7 •8 •9 •10 •Diversión •Entretenimient o •Poder •Qué •Dónde •Voluntad	•Juego •Nombre •Mejor •Lugar •Persona •Mente •Solo •Tú •Esto •Acción •Salir •Jugar •Yo •Ingresos •Motocicleta •Hola
Ambas	•Café •Pobre •Capacidad •Shock	•Encantado de conocer •Preguntar

Tabla 9 Conjunto de datos de 50 diferentes palabras en la lengua de signos coreana, incluidas en el conjunto de datos principal con respecto al lugar donde se realizan los movimientos (rostro, cuerpo o ambas), tomado de (Park, et al., 2021).

Una vez definido el conjunto de datos a utilizar, estos pasan a ser utilizados para entrenar y validar el modelo de red neuronal de SUGO. De acuerdo con Según Park, et al. (2021), en SUGO se adoptó un enfoque diferente al tradicional (combinación de redes neuronales convolucionales y redes neuronales recurrentes) y se utilizó una arquitectura de red neuronal convolucional en tres dimensiones para el procesamiento de los videos procesados.

Las redes neuronales convolucionales en tres dimensiones son una extensión de las redes neuronales convolucionales convencionales (en dos dimensiones, como las utilizadas en los ejemplos descritos hasta el momento), este tipo de red neuronal está diseñada para procesar datos volumétricos. Es por esta razón que se decidió implementar esta arquitectura como modelo para SUGO, la red neuronal convolucional en tres dimensiones extrae eficazmente los patrones temporales de los movimientos de la mano de un usuario a partir de marcos de imagen de profundidad, lo que permite una traducción precisa del lenguaje de señas (Park, et al., 2021).

Un aspecto importante para resaltar es que SUGO no es la única implementación de redes neuronales en tres dimensiones para la detección de gestos. Un estudio realizado por Al-Hammadi, et al. (2020) se propuso una arquitectura de redes neuronales convolucionales en tres dimensiones

adaptada para el reconocimiento automático de gestos con las manos, esta red neuronal fue diseñada para que su entrenamiento esté basado en las características espaciotemporales de las manos de las personas que realizan gestos en lengua de signos árabe y en lengua de signos estadounidense.

En el caso de Al-Hammadi, et al. (2020), los datos utilizados para el entrenamiento del modelo fueron obtenidos de tres fuentes, la primera de la Universidad King Saud, el cual comprende gestos seleccionados de las palabras y expresiones comunes de la lengua de señas saudita; la segunda de la Facultad de Ingeniería de la Universidad Americana de Sharjah, el cual contiene 150 muestras de 23 gestos diferentes realizados por diferentes personas repetidamente; y la última, de la Universidad de Purdue, el cual consiste en 43 gestos aislados con las manos en la lengua de signos estadounidense.

En cuanto al rendimiento de ambos modelos, Park, et al. (2021) evaluó el modelo implementado para SUGO en distintas condiciones lumínicas y realizando distintas combinaciones entre los sensores de los dispositivos móviles, como el uso de los valores RGB (rojo, verde y azul) de los píxeles de las imágenes obtenidas por los sensores ópticos utilizados, los valores en las tres dimensiones generadas por los sensores de profundidad y la combinación de ambos sensores. Esta comparativa se muestra en la Tabla 10, en la cual se muestra que el sensor de profundidad es el que mejores resultados produjo en las distintas condiciones en las que SUGO fue probado.

Condiciones del entorno	Ubicación seleccionada	Prueba con el conjunto de datos con valores RGB	Prueba con el conjunto de datos con valores RGB y profundidad	Prueba con el conjunto de datos con valores de profundidad
Interior	Oficina	82.89%	87.78%	91.31%
	Cuarto con alta iluminación	72.61%	87.56%	95.51%
	Cuarto con baja iluminación	64.08%	81.55%	90.08%
Exterior	Durante el día	67.83%	80.67%	92.19%
	Durante la noche	64.95%	83.51%	93.36%
	Durante la noche con iluminación callejera	74.63%	82.35%	90.69%

Tabla 10 Precisión de clasificación con respecto a diferentes condiciones de iluminación para los conjuntos de datos RGB, combinación de RGB y profundidad y sólo profundidad (Park, et al., 2021).

Por otra parte, en el estudio de Al-Hammadi, et al. (2020) se realizaron pruebas con cada uno de los conjuntos de datos seleccionados (para la lengua de signos árabe y la lengua de signos estadounidense), la comparativa de estos conjuntos de datos se realizó de dos formas, la primera desde un punto de vista

dependiente de la persona signante y la segunda de un modo independiente a la persona signante. Los resultados de ambos experimentos se muestran en la Tabla 11.

Modo	Precisión de la red neuronal para la detección de gestos en lengua de signos árabe y lengua de signos estadounidense		
	Conjunto de datos de la Universidad King Saud	Conjunto de datos de la Facultad de Ingeniería de la Universidad Americana de Sharjah	Conjunto de datos de la Universidad de Purdue
Precisión en modo dependiente del signante	98,12%	100,00%	76.67%
Precisión en modo independiente del signante	84,38%	34,90%	70,00%

Tabla 11 Comparativa de precisión de la red neuronal por cada conjunto de datos, separado por el modo de implementación (dependiente del signante o independiente del signante), basado en Al-Hammadi, et al. (2020).

Un punto importante resaltado en el estudio de Al-Hammadi, et al. (2020) es el del modo de implementación utilizado, ya que se logra ver una diferencia considerable en la precisión de los modelos según el modo probado. La diferencia entre un modo y otro es que, en el modo independiente del signante, el sistema debe estar capacitado para detectar gestos de personas que no forman partes del conjunto de datos de entrenamiento. Si se utiliza una nueva persona en la etapa de entrenamiento con el sistema independiente del signante, se obtiene un sistema adaptable capaz de aceptar a todos los nuevos signantes (Fakhfakh & Jemaa, 2018).

2.5 Desafíos en la detección de lengua de signos

La detección de gestos en lengua de signos trae consigo una serie de desafíos desde distintos puntos de vista, ya que existe una amplia variedad de aspectos a considerar al momento de realizar una traducción de los gestos de los signantes a través de una computadora. En esta sección se discuten los desafíos más importantes que se han encontrado durante el proceso de investigación.

2.5.1 Complejidad de la lengua de signos

El primer desafío al momento de querer detectar e interpretar gestos de los signantes es la lengua de signos como tal, al igual que los idiomas hablados, las lenguas de signos tienen una amplia variedad de gestos y expresiones que pueden ser utilizados en contextos totalmente diferentes. Cada signo hecho por el signante significa una letra, palabra o expresión distinta, la combinación de signos forma una oración al igual que las palabras en los idiomas hablados hacen oraciones; por lo tanto, una lengua de signos es una lengua natural completa con su propia sintaxis y gramática (Farooq, et al. 2021).

Cada lengua de signos varía en los gestos que forman parte de su vocabulario, en términos de las partes del cuerpo que se ven involucradas al momento de transmitir ideas. En la lengua de signos española, al igual que otras lenguas de este tipo, las partes del cuerpo que predominan son las manos y, dependiendo del gesto, se puede utilizar una o ambas manos, en el caso de que un gesto involucre ambas manos, “ésta puede ser simétrica, o bien, puede presentar una mano no activa, generalmente la izquierda, y la mano derecha con movimiento [...] en este caso, la articulación del signo sigue la condición dominante de la mano derecha” (González, 2003).

En el caso de la lengua de signos española, en esta se identifican “cambios en alguno de los parámetros articulatorios del signo producen [...], como el movimiento repetido para indicar el plural, o para expresar la duración de la acción; o bien, [...] para indicar si el sujeto es agente o paciente” (González, 2003). Todas estas variaciones en los gestos se deben tomar en consideración al momento de establecer los movimientos y partes del cuerpo que están involucradas en la elaboración de gestos en lengua de signos.

Por otra parte, al igual que cualquier otra lengua oral, la lengua de signos constantemente sufre cambios y en su estructura, debido a la evolución de las condiciones sociales y culturales de los signantes, por lo que se requiere estar constantemente actualizado sobre las nuevas estructuras gramaticales y signos con nuevos significados que puedan crearse a lo largo del tiempo. De acuerdo con Valdés-González y Martín-Antón (2020), “son más complicados de difundir y se divulgan, principalmente a través del contacto entre personas sordas y/u oyentes [...]. Aunque, gracias al acceso a Internet y el uso de filmaciones, la difusión de los nuevos signos es [...] más fácil”.

La ubicación en la que una lengua de signos se utiliza también puede afectar significativamente la estructura y significado de algunos gestos, similar a lo que ocurre con algunas palabras en las lenguas orales. Según Rodríguez (1992), la lengua de signos española presenta estas variantes “según países, según las ciudades, [...] según los colegios y según las Asociaciones de Sordos. Tales variantes afectan fundamentalmente al léxico gestual [...] El instrumento más frecuente de cohesión lingüística son los frecuentes intercambios culturales y deportivos que se llevan a cabo”.

De igual forma, así como sucede en las lenguas orales, cada individuo tiene su propia fluidez en la lengua de signos, de acuerdo con el nivel de conocimiento que tenga en esa lengua en específico. Un signante que tenga años de experiencia en la lengua de signos española o en cualquier otra lengua de signos tendrá una mejor fluidez que una persona que apenas esté aprendiendo a expresarse en la lengua de signos o tenga experiencia mínima en este, afectando el ritmo con el que expresa sus ideas hacia las demás personas.

Además de la fluidez de los individuos, el ritmo o velocidad por la cual se expresan las ideas varía de persona a persona, incluso el contexto en el que se entabla una conversación en lengua de signos afecta el ritmo en el cual se realizan los gestos, por ejemplo, la velocidad de una conversación casual entre amigos es distinta al ritmo de un intérprete de signos en un evento oficial. Según González (2003), “hay dos registros de uso del lenguaje de signos -formal e informal-, determinados por el nivel sociocultural de los participantes en la comunicación; ambos niveles son semejantes a los que se establecen en la lengua verbal”.

Es importante resaltar las diferencias entre la fluidez y la velocidad en la lengua de signos. La fluidez se refiere a la facilidad que tiene un individuo para expresar

sus ideas mediante gestos en lengua de signos, esto implica la capacidad de producir los signos de manera continua e ininterrumpida. Por otra parte, la velocidad se refiere al ritmo de palabras que un individuo tiene para elaborar gestos en lengua de signos, usualmente es una medida que toma en consideración la cantidad de gestos realizados en un periodo concreto de tiempo.

Todas estas características pueden influenciar en mayor o menor medida a la lengua de signos que se desea interpretar mediante técnicas de computación visual, inteligencia artificial o aprendizaje automatizado. Adeyanjua, et al. (2021) hace un énfasis en que actualmente existe la necesidad de profundizar los estudios para abordar el reconocimiento de signos con expresión facial, gestos con las manos y movimientos corporales simultáneamente con una mejor precisión de reconocimiento en tiempo real, los estudios prevén que estos desafíos se pueden lograr utilizando un enfoque de aprendizaje profundo con un sistema de alta configuración para procesar los datos de entrada con un tiempo computacional bajo.

2.5.2 Generación del conjunto de datos

Tal y como se ha definido en las secciones anteriores, cuando se trata de detectar gestos en cualquier lengua de signos, ya que el rendimiento de un modelo de aprendizaje automatizado depende fuertemente del conjunto de datos que se utilice para entrenar dicho modelo. Según Rastgoo, et al. (2021), con la llegada del aprendizaje profundo para el reconocimiento de la lengua de señas manual en los últimos años, se ha demostrado que tener una gran cantidad de datos es una parte crucial del aprendizaje del modelo.

Por otra parte, Farooq, et al. (2021) indica que los enfoques de aprendizaje profundo funcionan bien solo cuando se les proporciona un conjunto de datos de entrenamiento significativo y razonablemente inclusivo. Por lo anterior, es importante alimentar los modelos de aprendizaje automatizado con datos relevantes que ayuden a la generalización de los gestos en lengua de signos para un rendimiento idóneo.

Lo anterior no es un problema en el ámbito de traducción en tiempo real en lenguas orales, ya que existe una infinidad de conjuntos de datos en diversos lenguajes que ayudan a la creación de un conjunto de datos robusto; sin embargo, no existe esta diversidad de datos en el contexto de las lenguas de signos. Los conjuntos de datos públicos de lengua de signos tienen deficiencias que limitan el poder y la generalización de los sistemas entrenados con ellos, especialmente en el tamaño del conjunto de datos, la continuidad de los signos utilizados, el nivel de dominio de la lengua que poseen los signantes y la variedad que estos signantes pueden ofrecer al conjunto de datos.

El tamaño de los conjuntos de datos se puede ver influenciado por distintos factores, usualmente dependen del objetivo o enfoque al cual se le quiera dar al modelo de detección de gestos de lengua de signos. Los conjuntos de datos con gestos en lengua de signos, que son necesarios para impulsar el desarrollo de los sistemas de reconocimiento de gestos en este tipo de lenguas por lo general contienen menos de 100,000 gestos articulados (Bragg, et al., 2019).

Un ejemplo de estos conjuntos de datos es el desarrollado por el Centro de Normalización Lingüística de la Lengua de Signos Española (2023), el cual se denomina Corpus de la Lengua de Signos Española (CORLSE), siendo este “una recopilación de grabaciones en vídeo de una amplia variedad de discursos signados por personas sordas de distintas edades en diferentes lugares de España. Este corpus pretende ser una muestra representativa de la LSE para su consulta, documentación, estudio científico y uso para fines docentes”.

El CORLSE actualmente tiene la limitante de no contar con todos los gestos del vocabulario de la lengua de signos española, para contrarrestar este desafío, este conjunto de datos está diseñado para ser actualizado constantemente. De acuerdo con el último reporte publicado por el Centro de Normalización Lingüística de la Lengua de Signos Española (2023), esta colección de datos “cuenta ya con más de 2500 signos y es un recurso permanentemente abierto [...] Cualquier persona que quiera añadir un signo que utilice [...] lo podrá proponer para que el equipo de trabajo lo revise de cara a su inclusión”.

Asimismo, realizando una búsqueda en Kaggle, el cual es uno de los sitios más utilizados para la obtención de conjuntos de datos de toda índole, cuenta con un total de 446 conjuntos de datos a la fecha de este estudio al realizar una búsqueda con los términos “sign language”, estos conjuntos de datos varían desde algunas decenas de megabytes (conjuntos de datos pequeños) hasta decenas de gigabytes (conjuntos de datos grandes) la distribución de tamaños de acuerdo con la información de Kaggle se muestra la Ilustración 19.

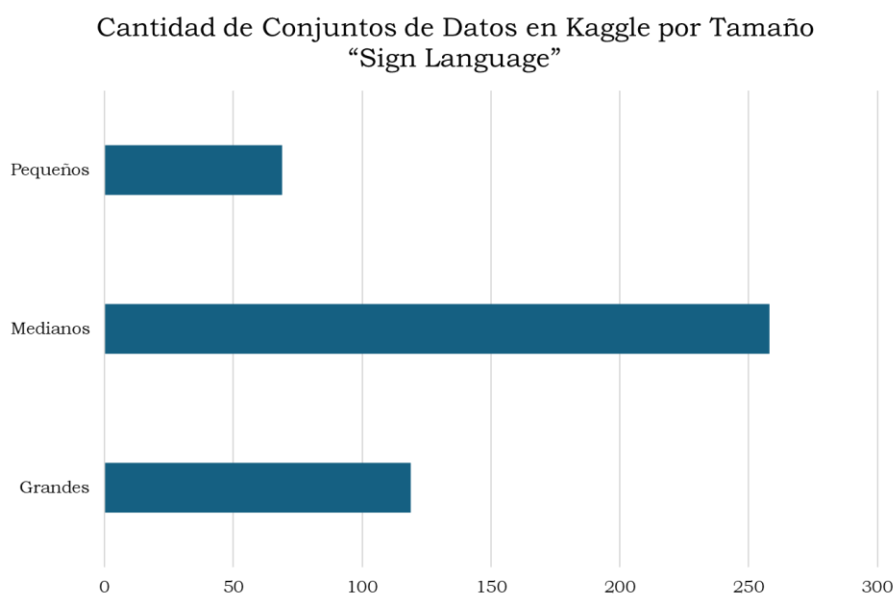


Ilustración 19 Distribución por tamaño de los conjuntos de datos en kaggle.com al realizar una búsqueda de los términos "sign language".

Además de la diferencia en el tamaño de los conjuntos de datos encontrados, la Ilustración 20 muestra los distintos formatos de los conjuntos de datos encontrados en Kaggle, ya que pueden dichas colecciones pueden contener imágenes estáticas de gestos específicos como letras del alfabeto, vídeos de articulaciones de palabras o expresiones complejas, secuencias que representan datos espaciales de los gestos recopilados, entre otras. Dado que un conjunto de datos puede tener una combinación de distintos tipos de archivos y formatos en lugar de limitarse a una estructura específica, los investigadores y profesionales que buscan información pública para desarrollar sus sistemas deben considerar este tipo de factores al momento de seleccionar un conjunto de datos para trabajar.

Formato de los Conjuntos de Datos Encontrados en Kaggle

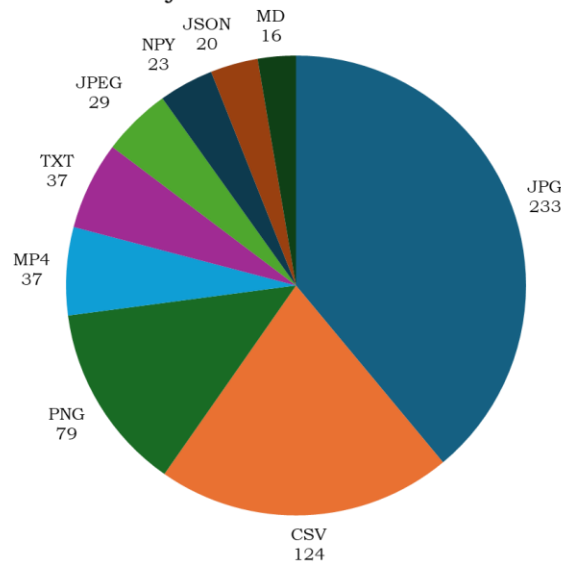


Ilustración 20 Formatos utilizados en los conjuntos de datos encontrados en kaggle.com cuando se realiza una búsqueda con los términos "sign language", como consideración, un conjunto de datos puede contener uno o más formatos en su colección.

De igual forma, dependiendo de los tipos de gestos en lengua de signos que se desean, los conjuntos de datos para el entrenamiento de los modelos de aprendizaje automatizado pueden contener tanto imágenes estáticas o secuencias de imágenes para mayor dinamismo. Muchos conjuntos de datos de gestos en lengua de signos existentes contienen gestos aislados que pueden ser importantes para ciertos escenarios (es decir, crear un diccionario de lenguaje de signos), sin embargo, la mayoría de los casos de uso del mundo real del procesamiento de la lengua de signos involucran conversaciones naturales con oraciones completas y expresiones más largas y complejas (Bragg, et al., 2019).

Un ejemplo común cuando se toma en cuenta la cantidad y tipo de gesto que se desea detectar mediante computadoras es el movimiento en los gestos. En la Ilustración 21 se muestra la diferencia entre la letra "R" y la letra "RR" del alfabeto de la lengua de signos española o la letra "I" y la letra "J" del alfabeto de la lengua de signos estadounidense. La posición de la mano es idéntica en ambos gestos, sin embargo la diferencia está en que en la "RR" y en la "J" hay un movimiento de la mano, por lo que pasa de ser una imagen estática a una secuencia de imágenes que componen el gesto completo.

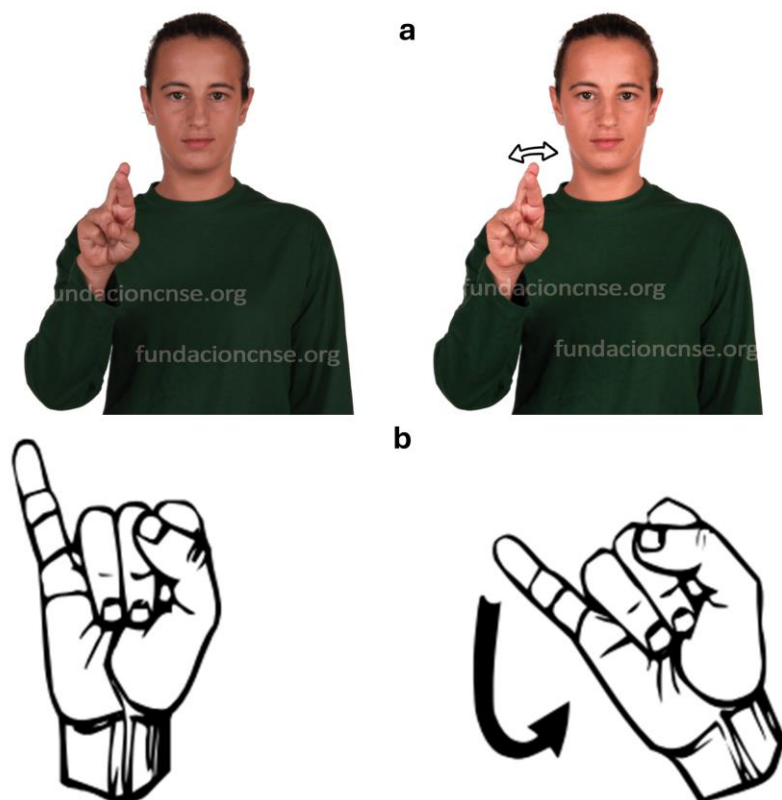


Ilustración 21 Gestos que representan a) las letras "R" y "RR" del alfabeto de la lengua de signos española y (Fundación de la Conferación Estatal de Personas Sordas, s.f.) b) las letras "T" y "J" del alfabeto de la lengua de signos estadounidense (Burbank Unified School District, s.f.).

De todos los 446 conjuntos de datos encontrados en Kaggle, uno de ellos contiene una colección de imágenes con gestos que representan la mayoría de las letras del alfabeto en la lengua de signos española, este conjunto de datos fue capturado en una pantalla blanca con una cámara de alta calidad con 3 manos diferentes a la misma distancia aproximada. La limitante con este tipo de conjunto de datos es que son representados únicamente a través de imágenes estáticas, por lo que cualquier otra letra que involucre el movimiento de las manos no será contada en los sistemas de detección que se elaboren basándose en estos datos.

Como se aprecia en la Ilustración 22, el conjunto de datos de Kaggle con el alfabeto de la lengua de signos española únicamente cuenta con 19 de los 28 posibles gestos para representar cada letra del alfabeto (tomando en consideración que las letras "LL" y "RR" tienen su propio gesto en este alfabeto), siendo los gestos de las letras "J", "LL", "Ñ", "R", "V", "W", "X", "Y" y "Z" excluidas de este conjunto de datos debido a que no son posibles de representar mediante imágenes estáticas que sirvan para el entrenamiento adecuado de un modelo de aprendizaje automatizado para el reconocimiento de gestos para la traducción de lengua de signos.

A (102 files)

Data Explorer
Version 1 (5.32 GB)

- fondo_blanco
 - A
 - B
 - C
 - D
 - E
 - F
 - G
 - I
 - K
 - L
 - M
 - N
 - O
 - P
 - Q
 - R
 - S
 - T
 - U

Ilustración 22 Captura del conjunto de datos publicado en Kaggle con imágenes estáticas del alfabeto de la lengua de signos española, en la cual se muestra que dicha colección de imágenes carece de las letras “J”, “LL”, “N”, “R”, “V”, “W”, “X”, “Y” y “Z” debido al involucramiento de movimiento de las manos para representar estas letras.

Otro aspecto importante para considerar es el nivel de experiencia que tenga un signante al momento de crear o utilizar un conjunto de datos para el reconocimiento de gestos en lengua de signos. Según Bragg, et al. (2019), esta diferencia en el dominio de los signantes cambia la ejecución al simplificar el estilo y el vocabulario, o haciendo signos más despacio para facilitar la comprensión.

El nivel de dominio de la lengua de signos de los signantes utilizados para la creación del conjunto de datos afecta directamente el rendimiento de los sistemas de detección de gestos, especialmente en aquellos sistemas que son dependientes del signante. Para Mohamed, et al. (2021), el reconocimiento de gestos con las manos es muy desafiante debido a la capacidad del sistema para manejar entradas que varían considerablemente de los datos utilizados durante la etapa de desarrollo, como ser la variabilidad de los signantes, la variabilidad del lenguaje, entre otros.

En contraparte con este desafío, un conjunto de datos en el cual se utilicen signantes de distintos niveles de dominio de la lengua de signos puede potencialmente aumentar el rendimiento de un sistema de detección de gestos, haciéndolo incluso independiente de signante. De acuerdo con Mohamed, et al. (2021), este tipo de sistemas son deseables para aplicaciones en el mundo real, ya que pueden ser utilizados por una amplia gama de usuarios sin necesidad de que los usuarios tengan un dominio pleno de la lengua de signos que se desea interpretar.

Bajo este mismo contexto, la velocidad mediante el cual un signante realiza los gestos en lengua de signos representa un desafío para la creación del conjunto de datos a utilizar para detectar gestos mediante computadora. De acuerdo con

Wang, et al. (2022), a través del análisis de las características del movimiento de la mano, se puede ver que el área de la mano es pequeña y la velocidad del movimiento de la mano es rápida, por lo tanto, es difícil extraer completamente la información del movimiento de la mano solo a través del reconocimiento global de imágenes.

La diversidad de los signantes para la creación de un conjunto de datos para lengua de signos no solamente se limita al nivel de dominio que tenga un individuo en una lengua de signos específica. De acuerdo con Bragg, et al. (2019), para reflejar con precisión los gestos en lengua de signos y escenarios de reconocimiento realistas, los conjuntos de datos deben incluir signantes que varíen en género, edad, ropa, geografía, cultura, tono de piel, proporciones corporales, discapacidad, fluidez, paisaje de fondo, condiciones de iluminación, calidad de la cámara, ángulos de cámara, entre otros.

Como se mencionó anteriormente, factores como la ropa y el fondo son importantes para la detección de gestos, especialmente si se desea crear un sistema de detección de gestos de lengua de signos mediante medios visuales. Para este tipo de sistemas usualmente se emplean técnicas y algoritmos de computación visual, los cuales toman en cuenta cada píxel obtenido por la cámara de vídeo, por lo que se necesita un contraste suficiente entre el signante y el fondo que se está utilizando.

En la Ilustración 23 se muestra una captura de dos signantes del Centro de Normalización Lingüística de la Lengua de Signos Española, este ejemplo es de utilidad para identificar visualmente cómo un algoritmo de computación visual tendría más dificultad para detectar a la persona signante que se encuentra a la izquierda (un fondo azul oscuro con una vestimenta de un tono ligeramente más oscuro, lo cual se representa en valores RGB similares en los píxeles de la imagen) en comparación con la persona signante de la derecha (un fondo azul oscuro con una vestimenta de un color distinto y de alto contraste como es el gris claro, lo cual facilitaría distinguir entre los valores RGB de los píxeles que representan el fondo y los valores RGB que representan a la signante como tal).

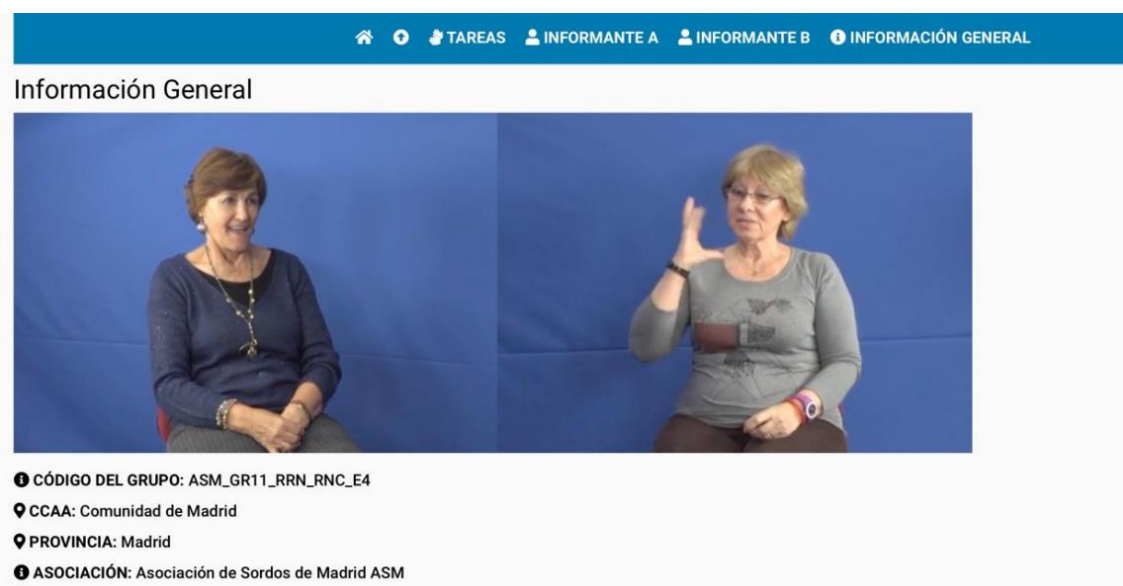


Ilustración 23 Captura del sitio del Corpus de la Lengua de Signos Española (CORLSE), en el cual se muestra a dos signantes con distintos niveles de contraste entre el fondo utilizado y la vestimenta de los individuos (Centro de Normalización Lingüística de la Lengua de Signos Española, 2023).

En el caso de los sistemas de detección de gestos basados en cámaras de vídeo existe una amplia variedad en el formato utilizado para la generación del conjunto de datos. De acuerdo con Bragg, et al. (2019), los conjuntos de datos de lenguaje de signos existentes suelen consistir en vídeos de personas haciendo gestos en lengua de signos, el formato de estos vídeos puede variar de acuerdo con el dispositivo de grabación, por ejemplo, las cámaras de vídeo a menudo producen formato MP4, OGG, AVI, entre otros.

Por otra parte, referente a los sistemas de detección de gestos en lengua de signos utilizando dispositivos físicos como guantes también presentan una amplia variedad de formatos y aplicaciones. Los conjuntos de datos de captura de movimiento se han seleccionado, a menudo mediante la conexión de sensores a un signante; estos pueden ser utilizados para generar avatares y animaciones para representar gestos en lengua de signos; también los dispositivos como el Microsoft Kinect pueden ser utilizados para capturar el posicionamiento en 3D y generar un conjunto de datos con la información recopilada (Bragg, et al., 2019).

2.6 Técnicas para la detección de gestos en lengua de signos

Tal y como se ha descrito en este estudio, existe una amplia variedad de técnicas a utilizar, sin embargo, no todas tienen las capacidades necesarias para la traducción en tiempo real de las lenguas de signos. Núñez-Marcos, et al. (2023) indica que la comunicación entre un signante y un individuo que no tenga conocimiento de la lengua de signos debe ser inmediata, de lo contrario no será apropiada para situaciones del mundo real; si el tiempo para inferir una traducción es muy largo, los usuarios pueden perder interés en la aplicación y abandonarla rápidamente.

De igual forma, cada técnica empleada tiene sus ventajas y desventajas en ciertos aspectos. En el caso de la detección de gestos mediante guantes, estos tienen la desventaja de ser intrusivos por su naturaleza de ser dispositivos que los signantes deben utilizar obligatoriamente para la detección del movimiento. El uso de guantes para la detección de gestos suele ser costoso y, sobre todo, intrusivos para los usuarios, las personas signantes no se sentían cómodas con este enfoque, dado que necesitaban usarlas para poder comunicarse (Núñez-Marcos, et al., 2023).

Otro aspecto para considerar de los guantes es que, al contar con componentes electrónicos, estos pueden ser susceptibles a aumentos en la temperatura, lo cual si no se regula debidamente puede ocasionar un sobrecalentamiento en los guantes. Las aplicaciones basadas en guantes están aumentando en todos los campos, como la robótica, la medicina, la mecánica y los juegos; sin embargo, es necesario tomar en consideración el uso de guantes resistentes al calor (Harish & Poonguzhali, 2022).

Como alternativa a los dispositivos de detección de gestos invasivos como los guantes, se puede optar por utilizar técnicas visuales como las discutidas a lo largo de esta investigación, tomando en consideración que este enfoque también tiene sus limitantes. Bragg, et al. (2019) indica que, a pesar de las grandes mejoras de los últimos años, todavía hay muchos problemas de reconocimiento importantes y sin resolver, que dificultan la aplicabilidad en el mundo real, especialmente en la representación, el etiquetado y la generalización de los modelos de aprendizaje automatizado.

La representación de los gestos mediante técnicas de detección visuales depende de diversos factores, como la iluminación, contraste entre el signante y el fondo, entorno, entre otros. Para Pan, et al. (2016), los enfoques visuales para la detección de gestos en lengua de signos son más convenientes para los usuarios, ya que la cámara está disponible en casi todos los dispositivos móviles y no se necesita ningún sensor adicional; sin embargo, estos enfoques no funcionan bien si hay colores, texturas o movimientos complejos en el entorno de fondo.

Asimismo, debido a la complejidad de las estructuras gramaticales en las lenguas de signos, un término puede contar con múltiples interpretaciones en gestos, o un gesto puede representar múltiples términos dependiendo del contexto en el cual estas ideas se expresan. Las técnicas actuales no pueden manejar representaciones de estos términos, múltiples representaciones pueden expresar el mismo concepto, y los sistemas no tienen una forma estándar de codificar esta riqueza en la lengua de signos (Bragg, et al., 2019).

En el mismo contexto de la riqueza de las lenguas de signos, otro desafío que se debe tomar en cuenta en los modelos de aprendizaje automatizado para la interpretación de gestos es el etiquetado de los elementos del conjunto de datos. Este etiquetado corresponde al proceso de marcar datos específicos en un vídeo o conjunto de datos para indicar la presencia y las características de los gestos de la lengua de signos, como ser la posición de los dedos de las manos de los signantes, expresiones faciales, entre otros.

Estas etiquetas son importantes para el proceso de entrenamiento y verificación del rendimiento de modelos que forman parte de los sistemas de reconocimiento de lenguaje de señas. De acuerdo con Bragg, et al. (2019), no existe un sistema de etiquetado estandarizado ni un nivel de granularidad en las etiquetas de los conjuntos de datos para la detección de lengua de signos, esto impide a los investigadores combinar conjuntos de datos etiquetados para aumentar la potencia de los modelos que desarrollan.

Finalmente, uno de los desafíos principales en los modelos de aprendizaje automatizado de cualquier índole es la generalización de los resultados obtenidos. Bajo este contexto, la generalización se puede considerar como la capacidad de un modelo entrenado para aplicar lo que ha aprendido de los datos de entrenamiento a datos nuevos que son distintos al conjunto de datos de entrenamiento original. La generalización es una de las principales dificultades del aprendizaje automatizado, y los sistemas de reconocimiento de la lengua de signos no es una excepción; se necesita de conjuntos de datos más grandes y diversos para entrenar modelos que logren obtener esta generalización en sus predicciones (Bragg, et al., 2019).

El desafío que afronta el desarrollo de sistemas de detección de gestos en lengua de signos inicia desde la generación del conjunto de datos para el entrenamiento y verificación. Según Mohamed, et al. (2021) La mayoría de los conjuntos de datos disponibles para el público en general se han elaborado en condiciones de laboratorio con fines de investigación lingüística, estos dependen del individuo que elabora los gestos. Debido a esto, cuando se entrena un modelo de aprendizaje automatizado con este tipo de conjunto de datos, no tiene un alto grado de generalización, ya que las estructuras gramaticales utilizadas ya se encuentran diseñadas con anterioridad, sin contar variaciones como las descritas en secciones anteriores de este estudio.

2.6.1 Limitaciones de los dispositivos para la detección de gestos en lengua de signos

Dependiendo del enfoque que se desee tomar al momento de desarrollar un sistema de detección de gestos en lengua de signos, se utilizará un tipo de hardware u otro. Esta diversidad predomina en mayor medida en las técnicas de detección por medios visuales, ya que, como se ha discutido en secciones anteriores, se pueden usar una amplia gama de sensores para la detección de las partes del cuerpo que se utilizan para hacer gestos en lengua de signos, entre ellos cámaras de vídeo, sensores de movimiento, cámaras estereoscópicas, sensores infrarrojos, entre otros.

El desafío del hardware a utilizar tiene un mayor realce cuando se desea aplicar este tipo de sistemas a dispositivos móviles. En el caso de SUGO, el cual se basa en el uso de los sensores de los dispositivos móviles para la detección de vídeo y profundidad, su funcionalidad dependerá en gran medida del tipo de teléfono móvil que se utilice. De acuerdo con Park, et al. (2021), las cámaras de profundidad de los teléfonos inteligentes tienen varias limitaciones fundamentales para la traducción del lenguaje de señas, específicamente, los sensores de profundidad basados en infrarrojos como el Apple TrueDepth de los iPhones.

El mayor inconveniente que muestran estos sensores para la detección de profundidad es que pueden verse afectados en su rendimiento cuando la iluminación es muy alta. La Ilustración 24 describe gráficamente la diferencia que existe cuando el entorno cambia drásticamente, ya que los datos recopilados en el exterior durante un día soleado muestran bordes menos claros en comparación con los datos recopilados en un entorno interior, lo cual dificulta la tarea de detección de gestos a través de sensores infrarrojos.



(a) Outdoor sunny environment



(b) Indoor office environment

Ilustración 24 Comparativa de imágenes de profundidad capturadas en con la tecnología TrueDepth de Apple en a) el exterior con un día soleado y b) entornos interiores (Park, et al., 2021).

De igual forma, no todos los modelos de teléfonos inteligentes tienen sensores de profundidad incorporados, factores como la gama del teléfono, el modelo o las decisiones que los fabricantes tomen referentes al diseño del teléfono influyen en los tipos de sensores que estos dispositivos tendrán para el uso de las personas.

Un problema similar afronta los sistemas de detección de gestos en lengua de signos que utilicen cámaras estereográficas para la detección visual de gestos.

Por una parte, se puede utilizar una cámara estereográfica dedicada para la detección de gestos. Muchas de ellas vienen incorporadas de fábrica en dispositivos de realidad aumentada, sin embargo, esta opción puede resultar invasiva para algunos usuarios. Por lo anterior, una alternativa es el uso de las distintas cámaras que tienen los dispositivos móviles para hacerlas funcionar como una cámara estereográfica.

En la actualidad la gran mayoría de teléfonos inteligentes cuentan con al menos dos cámaras, las cuales podrían utilizarse en conjunto para simular el comportamiento de una cámara estereográfica. De acuerdo con Park, et al. (2021), desde el iPhone X, Apple ha incluido un sensor TrueDepth para la captura de profundidad, mientras que teléfonos inteligentes como el Pixel 4 ofrece capacidades de captura de profundidad utilizando múltiples cámaras.

Como resultado de la explotación de los recursos de los teléfonos móviles, sistemas de detección de gestos en dispositivos móviles como SUGO son capaces de realizar todas las operaciones necesarias para la traducción completamente en el teléfono inteligente, lo que minimiza la latencia de inferencia y reduce las posibles invasiones de privacidad debido al intercambio continuo de imágenes a través de la red inalámbrica (Park, et al., 2021).

3 Desarrollo

A continuación, se describe en detalle la implementación del sistema propuesto para la de detección de gestos en tiempo real para la lengua de signos española mediante la segmentación e identificación de gestos a través de una cámara web.

3.1 Especificación de hardware utilizado

El desarrollo del sistema de detección de gestos en lengua de signos española se llevó a cabo en un ordenador portátil personal. Este dispositivo sirvió como plataforma principal para codificar, verificar y depurar el sistema, lo que garantizó que todas las funcionalidades se integraran de forma idónea.

Utilizando los recursos y el software disponibles se ha podido implementar y refinar el sistema de manera eficiente, superando desafíos y optimizando el rendimiento de este. La portabilidad y la comodidad del uso de un portátil permitió un desarrollo continuo en diversos entornos, lo que contribuyó significativamente a la finalización exitosa del proyecto. Las especificaciones de hardware y software del ordenador portátil utilizado se describen en la siguiente tabla:

Especificación	Descripción
Sistema operativo	Microsoft Windows 11 Home Edition de 64 bits
Versión de sistema operativo	23H2
Fabricante del ordenador	Lenovo, Group Limited
Modelo del ordenador	IdeaPad Gaming 3 - 15ARH7 - 82SB
Fabricante del procesador	Advanced Micro Devices (AMD)
Modelo del procesador	Ryzen 5 6600H con AMD Radeon Graphics, 3301 Mhz, 6 núcleos, 12 procesadores lógicos
Arquitectura del procesador	64 bits
Tamaño de memoria RAM	8 Gb - DDR5
Almacenamiento interno	256 Gb
Tipo de procesador gráfico	Combinado (dedicada e integrada)
Fabricante del procesador gráfico	Nvidia Corporation
Modelo del procesador gráfico	GeForce RTX 3050 Laptop GPU
Memoria RAM del procesador gráfico	8 Gb - DDR4
Tipo de cámara	Cámara web integrada
Resolución de la cámara	0.9 megapíxeles – 1280 x 720

Tabla 12 Especificaciones de hardware y software del ordenador portátil utilizado para la codificación, verificación y depuración de la solución propuesta para el sistema de detección de gestos en lengua de signos.

Referente a la Tabla 12, es de resaltar que el uso de un procesador de vídeo dedicado es de gran utilidad al momento de desarrollar este tipo de soluciones que se basan en modelos de aprendizaje automatizado. Usualmente el proceso de entrenamiento de los modelos conlleva una serie de operaciones matemáticas

que requieren de una capacidad de cómputo superior si se desea realizar este proceso de forma eficiente.

Por lo general, la tarea de realizar los cálculos matemáticos es realizada por los núcleos del procesador, sin embargo, las librerías que se utilizan para la creación de modelos de inteligencia artificial están diseñadas para aprovechar la capacidad de cómputo que tienen los procesadores gráficos para realizar cálculos de forma paralela al procesador principal.

3.2 Elaboración del conjunto de datos

El conjunto de datos utilizado en este estudio consiste en una combinación de recursos obtenidos de internet y recursos de autoría propia. La razón por la cual se ha decidido tomar este enfoque es debido a la falta de diversidad en los conjuntos de datos en la lengua de signos española. Si bien existen diccionarios disponibles en internet, algunos tienen ciertas limitaciones, como la baja complejidad de las expresiones existentes o las restricciones en el uso de esta información para este estudio.

Como se mencionó en la sección 2.5.2 Generación del conjunto de datos, uno de los conjuntos de datos identificados para la lengua de signos española en Kaggle consiste en una colección de imágenes estáticas que representan la mayoría de las letras del alfabeto español. Sin embargo, este conjunto de datos no puede ser utilizado para este estudio debido a que se requiere analizar el movimiento continuo del cuerpo de los signantes para realizar una traducción de los gestos en lengua de signos española en tiempo real.

Por otra parte existen recursos en línea que cuentan con videos que muestran gestos en lengua de signos española para distintos términos en español. Si bien esta alternativa es la que mayor potencial presenta, la información encontrada para expresiones de baja, media y alta complejidad varía entre los sitios. En la Tabla 13 se listan los sitios encontrados durante el proceso de investigación y las características positivas y negativas de cada sitio.

Nombre del sitio	Dirección web	Ventajas	Desventajas
Diccionario de la Lengua de Signos Española	fundacioncnse-dilse.org	Este es un sitio elaborado por la Confederación Estatal de Personas Sordas que cuenta con una colección de aproximadamente 7.728 expresiones de la lengua de signos española, asimismo, cuenta además con videos en alta definición.	Los términos encontrados en esta página son en su mayoría palabras aisladas y no expresiones más complejas
Spread the Sign	spreadthesign.com	Este sitio está realizado por una organización internacional con gestos de expresiones en distintas lenguas	Esta colección es muy completa para la lengua de signos española, cada expresión

Nombre del sitio	Dirección web	Ventajas	Desventajas
		de signos, también cuenta con expresiones con estructuras gramaticales más complejas.	cuenta con un máximo de 3 signantes y de todas las opciones es la que ofrece vídeos con la resolución más baja.
Corpus de la Lengua de Signos Española (CORNLE)	corpuslse.es	Esta es una colección de vídeos de personas sordas realizando discursos entre ellos, a la fecha de este estudio el sitio tiene un total de 98 grabaciones con expresiones de alta complejidad y en vídeos de alta definición.	Las expresiones utilizadas en este repositorio son muy complejas para utilizarlas en el conjunto de datos, ya que contiene secuencias muy grandes de gestos.
Sematos	sematos.eu	Este es un sitio creado para almacenar gestos en distintas lenguas de signos con una cantidad de aproximadamente 6.076 expresiones en la lengua de signos española	Este sitio no permite la descarga de los recursos multimedia para su uso en el conjunto de datos.

Tabla 13 Cuadro comparativo con las características positivas y limitaciones de los sitios web con diccionarios de lengua de signos española.

Tomando en consideración lo descrito en la tabla anterior, el conjunto de datos generado para este estudio está compuesto por vídeos de autoría propia, de Spread the Sign y del Diccionario de la Lengua de Signos Española. Para cada término que forma parte del vocabulario seleccionado para el conjunto de datos, se generaron 30 secuencias distintas.

Dado que ninguno de los sitios encontrados en internet cuenta con la variedad de vídeos por cada expresión utilizada para este estudio, los 30 vídeos por término fueron complementados con vídeos de autoría propia utilizando la cámara web del ordenador portátil utilizado para este trabajo de investigación. La distribución de los orígenes por palabra utilizada se muestra en la Tabla 14.

No.	Término	STS	DILSE	Webcam	Total
1	Bienvenido	1	1	28	30
2	Encantado de conocerte	1	1	28	30
3	Gracias	2	0	28	30
4	Hola	2	1	27	30
5	Madre	2	1	27	30
6	Padre	3	1	26	30
7	Saludo	2	1	27	30
	Total	13	6	191	210

Tabla 14 Lista del vocabulario utilizado para el entrenamiento de la red neuronal para la detección de gestos en lengua de signos española.

El procesamiento de los vídeos es en mayor parte similar para los obtenidos en internet como para los obtenidos a través de la cámara web, los fotogramas obtenidos por ambos orígenes son procesados a través de un código realizado en Python para obtener los vectores que representan distintos puntos de las manos, el torso y el rostro. En la Ilustración 25 se define gráficamente el proceso mediante el cual se generó el conjunto de datos para este estudio tomando los vídeos descargados de internet y los obtenidos desde la cámara web del ordenador portátil.

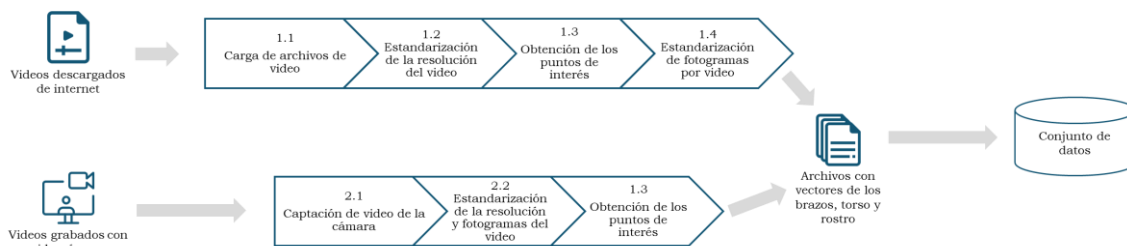


Ilustración 25 Proceso de creación del conjunto de datos utilizado para el estudio.

Tanto en los vídeos grabados con la cámara web del ordenador portátil como en los vídeos obtenidos de internet, decidí optar por las librerías OpenCV y MediaPipe de Python para el procesamiento de los vídeos, OpenCV se utiliza en mayor parte para recopilar los fotogramas de la cámara web y poder mostrar en pantalla mensajes que ayuden al proceso de recopilación de información para el conjunto de datos. MediaPipe, por otra parte, es utilizado para obtener las coordenadas de los puntos que representa la mano derecha, mano izquierda, torso y rostro de la persona que se encuentre en la toma. La información recolectada de los puntos de las partes del cuerpo es almacenada como archivos “.npz” que son interpretados por la librería Numpy para el manejo de arreglos y vectores en Python.

Para los vídeos obtenidos desde la cámara web del ordenador portátil, se utilizó OpenCV para obtener los fotogramas de la cámara, esta librería por defecto abre una ventana de 640x480 píxeles de tamaño. Una vez definida la resolución de los vídeos de muestra, utilicé MediaPipe para obtener los puntos de interés (landmarks) de las manos, torso y rostro, estos son almacenados en archivos “.npz” con la secuencia de coordenadas de las manos, torso y rostro del signante.

De forma similar, para los videos obtenidos de internet se elaboró un script en Python para procesar y transformar los fotogramas obtenidos hacia una colección de archivos que contienen los vectores que representan. Para lograr esto, primero se estandariza la resolución del video a la misma que la definida para los videos desde la cámara web del ordenador portátil (640x480 pixeles). Una vez estandarizada la resolución, se quitaron algunos fotogramas de cada video de forma dinámica, ya que la cantidad de fotogramas para el conjunto de datos es fija, mientras que cada video tiene una cantidad de fotogramas propia. Finalmente, se obtienen puntos de interés (landmarks) que representan las manos, torso y rostro del signante y se almacenan en un archivo “.npz” forma idéntica a los generados para los gestos tomados desde la cámara web.

La obtención de los puntos de interés (landmarks) se realizó mediante la librería MediaPipe, cuya función fue descrita en la sección “2.3.1.1. Uso de Cámaras de Video Convencionales para la Detección de Gestos”. Los datos obtenidos de MediaPipe corresponden a las coordenadas X, Y y Z que representan la posición en las que se encuentra cada punto (landmark) detectado por la librería. Visualmente estos puntos se pueden identificar en el resultado obtenido mediante OpenCV y que se muestra en la Ilustración 26.



Ilustración 26 Representación visual de los puntos de interés obtenidos con MediaPipe para la expresión “gracias” de a) los videos capturados desde la cámara web y b) los videos descargados de internet.

Por cada fotograma de los videos recopilados, se guarda un archivo “.npz” que contiene un arreglo con todas las coordenadas de los puntos de interés (landmarks), la distribución de los elementos en este arreglo se muestra en la Tabla 15.

Parte del cuerpo	Cantidad de landmarks	Dimensiones
Rostro	1.404	X, Y, Z
Pose	132	X, Y, Z, Visibilidad
Mano derecha	63	X, Y, Z
Mano izquierda	63	X, Y, Z
Total	1.662	

Tabla 15 Distribución de elementos en el arreglo guardado en cada archivo “.npz” con las posiciones de los puntos de interés (landmarks) obtenidos con MediaPipe.

Para cada gesto se guardaron 30 secuencias de 30 fotogramas cada una, lo que representan 900 archivos por cada expresión del vocabulario definido. Considerando que cada archivo tiene 1.662 elementos, cada expresión tiene 1.495.800 elementos, dando un total del conjunto de datos de 10.470.600 valores que representan las posiciones de las partes del cuerpo de los signantes.

3.3 Implementación de solución

Una vez definido y creado el conjunto de datos que se utilizará para este estudio, se necesita crear un programa para tomar este conjunto de datos y utilizarlo para entrenar un algoritmo de aprendizaje automatizado que se encargue de la detección de partes del cuerpo y la interpretación de los gestos expresados en lengua de signos.

Existen múltiples soluciones de uso público en internet para este propósito. Sin embargo, una gran cantidad de estos se enfoca en la detección de imágenes estáticas para la detección de gestos, lo cual no es parte de los objetivos de este estudio. Para lo cual se hizo un filtrado de la búsqueda utilizando algoritmos con redes neuronales recurrentes. Uno de los ejemplos encontrados es el de Renotte (2021), el cual es un repositorio público en línea una implementación práctica de la estimación de gestos en lengua de signos utilizando una red neuronal de corto plazo extendida construida en Python. Este fue el repositorio mediante el cual se basó el desarrollo de este estudio.

Para el diseño de la solución, se optó por una combinación entre redes neuronales de memoria a corto plazo extendidas y redes neuronales densas, la diferencia entre una y otra es que en la densa todos los nodos de la red están conectados con el resto de los nodos de la siguiente capa. El diseño de la red neuronal se muestra en la Ilustración 27.

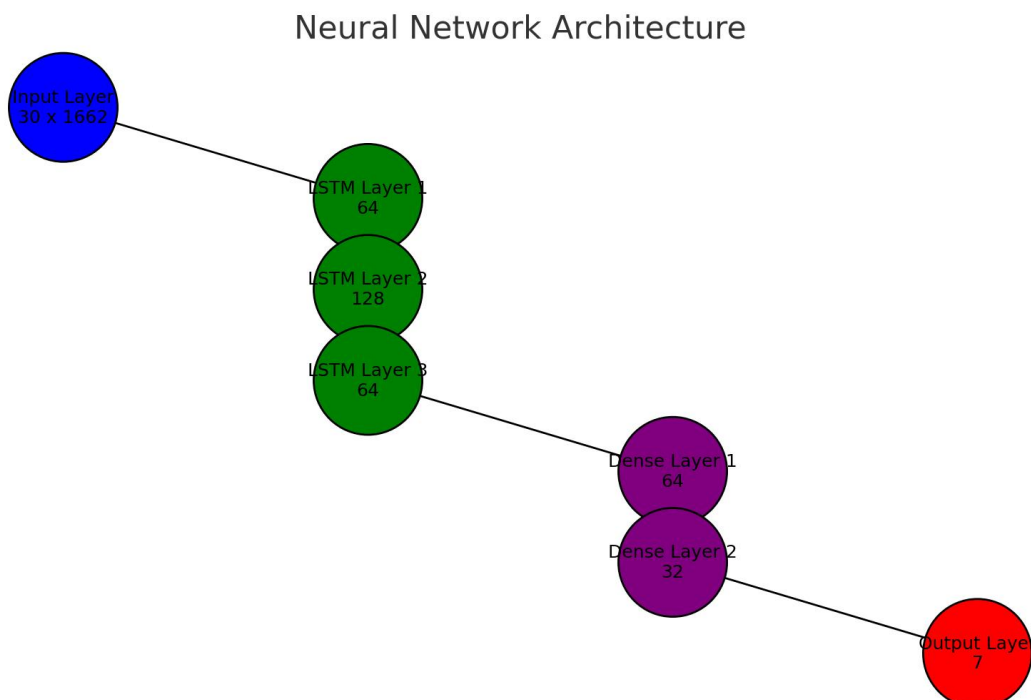


Ilustración 27 Diseño de la red neuronal utilizada para la detección de gestos en lengua de signos española a través de LSTM y redes neuronales densas.

Para la capa de entrada, el tamaño de entrada es (30, 1.662), ya que cada de secuencia de tiene una longitud de 30 (correspondiente a los 30 fotogramas por

secuencia) y cada elemento es un vector de 1.662 dimensiones, que corresponden a las coordenadas de los puntos de interés (landmarks).

Las primeras tres capas ocultas de la red neuronal tienen una configuración LSTM, ya que son las redes neuronales recurrentes que procesarán la secuencia de fotogramas a lo largo del tiempo. Las siguientes dos capas de la red son redes neuronales densas, y estas están conectadas a la capa de salida. Como el conjunto de datos seleccionado consta de 7 palabras, la capa de salida consta de 7 nodos, cada uno representando un posible resultado en la predicción de la red. La configuración de cada nodo se muestra en la Tabla 16.

Capa	Tipo de cada	Tamaño de nodos
Entrada	Nodos de entrada	(30, 1.662)
Capa oculta 1	LSTM	64
Capa oculta 2	LSTM	128
Capa oculta 3	LSTM	64
Capa oculta 4	Dense	64
Capa oculta 5	Dense	31
Capa de salida	Dense	7

Tabla 16 Descripción de las capas definidas para la red neuronal utilizada para la detección de gestos en lengua de signos.

Finalmente, para la optimización de la red neuronal se optó por el optimizador de estimación de momento adaptativo (Adam Optimizer), ya que este es un método popular utilizado para ajustar los pesos de una red neuronal durante el proceso de entrenamiento para minimizar los errores, dicho error se basa en una configuración de entropía cruzada categórica, ya que esta red neuronal resuelve un problema de clasificación múltiple (siete posibles soluciones).

El proceso de entrenamiento se realizó con el conjunto de datos generado para este estudio. La proporción de cantidad de datos de entrenamiento y prueba seleccionada fue de 90:10, es decir, 90 por ciento de los datos del conjunto fueron utilizados para el entrenamiento y el 10 por ciento fue destinado a la realización de pruebas para verificar la precisión de la red neuronal. La proporción seleccionada para este estudio permite provechar la mayor cantidad posible de datos, esto se debe a que, si bien se cuenta con una cantidad considerable de información, el volumen de datos y la diversidad de la información no es lo suficientemente alta para tomar otras distribuciones como la del principio de Pareto (80:20), que es una de las más utilizadas para la separación de conjunto de datos.

Se realizaron tres entrenamientos con 300, 2.000 y 500 épocas cada uno, obteniendo los mejores resultados con 500 épocas de entrenamiento. Esto se muestra de mejor forma en la Ilustración 27, aproximadamente desde la época 1.700 del proceso de entrenamiento la precisión de la red neuronal disminuye drásticamente, este fenómeno es común en el aprendizaje automatizado debido al sobre entrenamiento del modelo, lo cual se produce cuando el modelo aprende demasiado bien los datos de entrenamiento en lugar de capturar los patrones subyacentes. Como resultado, el modelo funciona muy bien en los datos de entrenamiento, sin embargo, con datos nuevos para el modelo la precisión baja considerablemente.

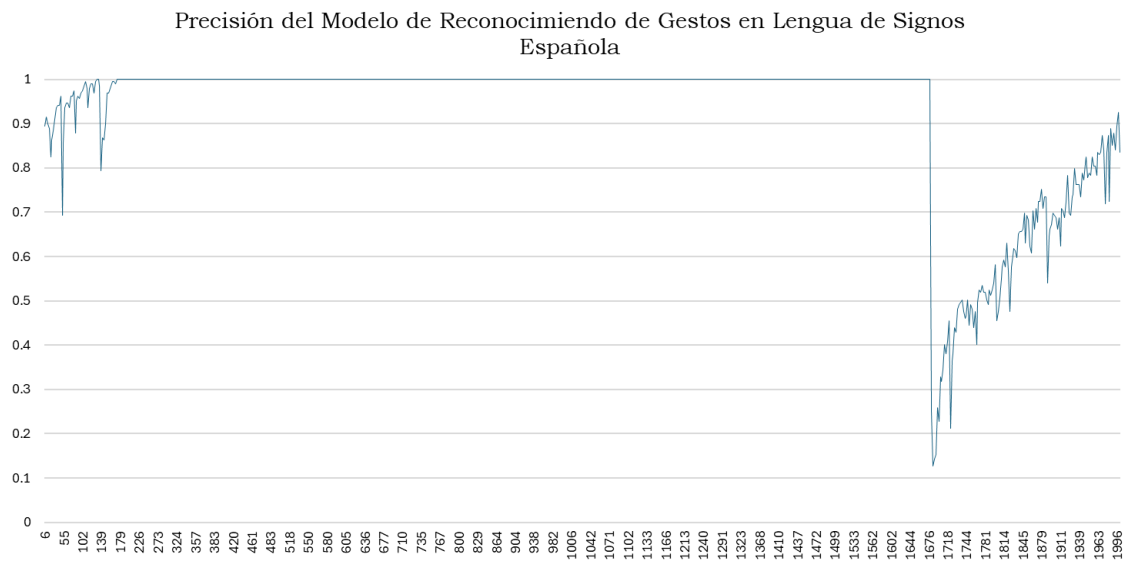


Ilustración 28 Evolución de la precisión del modelo desarrollado al momento de entrenarlo mediante 2.000 épocas.

Una vez entrenado el modelo de red neuronal diseñado, se utilizó la librería OpenCV de Python para abrir una ventana con la captura de fotogramas a través de la cámara web del ordenador portátil, estos a su vez capturan los puntos de interés con MediaPipe y utiliza el vector de puntos como entrada para predecir nuevos valores. Los resultados se describen en la siguiente sección.

4 Resultados obtenidos

Una vez desarrollada la solución, se realizaron pruebas para la detección de gestos en tiempo real del modelo. Para esto se hizo un script en Python que tomara los fotogramas desde la cámara web del ordenador portátil y capturara mediante la librería MediaPipe los puntos de interés de las partes del cuerpo del signante en cámara, una vez obtenido el vector de puntos de interés se envía esta información a la red neuronal para que hiciera una predicción basada en la información obtenida.

Los resultados se van mostrando en tiempo real a medida se van realizando gestos, la red neuronal devuelve una colección de siete valores, cada uno representando la probabilidad calculada por el modelo de que cierto término coincida con la información captada por la cámara web del ordenador. Como se mencionó en la sección anterior, se hicieron tres ciclos de entrenamiento con 200, 2.000 y 500 épocas, respectivamente; la Ilustración 29 muestra la diferencia en la precisión de los modelos en cada uno de los ciclos ejecutados, siendo el ciclo 3 el que mejores resultados arrojó, obteniendo una precisión promedio del 97.72 por ciento.

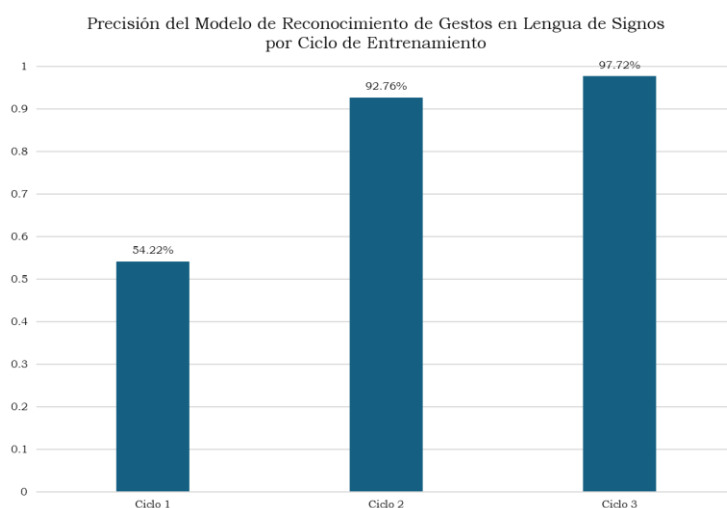


Ilustración 29 Precisión del modelo desarrollado utilizando 200 épocas (ciclo 1), 2.000 épocas (ciclo 2) y 500 épocas (ciclo 3).

La correlación entre el arreglo de salida de la red neuronal y los términos del vocabulario son los descritos en la Tabla 17.

Posición	Término del conjunto de datos
0	Bienvenido
1	Encantado de conocerte
2	Gracias
3	Hola
4	Madre
5	Padre
6	Saludo

Tabla 17 Correlación de los ítems en el arreglo obtenido de la red neuronal desarrollada y los términos utilizados en el conjunto de datos.

Esta correlación en los términos se muestra en la Ilustración 30, en la cual se está ejecutando el programa de reconocimiento y a medida se van realizando gestos se obtiene por parte de la red neuronal un posible resultado de lo que ese gesto puede significar de acuerdo con conjunto de datos que se utilizó para el entrenamiento, esto se ve representado con barras de color azul en cada término que selecciona el modelo de la red neuronal.



Ilustración 30 Detecciones en tiempo real realizadas por la red neuronal desarrollada para este estudio.

Un aspecto importante a mencionar, es que tanto para la generación del conjunto de datos como para las pruebas de detección en tiempo real decidí aplicar un contraste bajo entre la vestimenta del signante y el fondo utilizado, esto con el propósito de demostrar la versatilidad de librerías como MediaPipe que dependen únicamente de la detección de las partes del cuerpo que aparecen en cámara, sin importar los colores de fondo o de la vestimenta utilizada, lo cual ayuda a que este tipo de sistemas sea más preciso en cualquier tipo de escenarios.

5 Conclusiones

Este estudio tuvo la finalidad de desarrollar el prototipo de un sistema de reconocimiento de gestos mediante técnicas de aprendizaje automatizado y computación visual para la traducción de la lengua de signos española a texto. Los resultados obtenidos con el sistema desarrollado muestran un gran potencial para la traducción en tiempo real de gestos en lengua de signos española, ya que se obtuvo una precisión acorde a modelos similares desarrollados con técnicas similares en otras lenguas de signos y al mismo tiempo no requiere de una capacidad excepcional de cómputo para la realización de esta tarea.

En la última década el estudio de la detección de gestos en lengua de signos muestra un crecimiento constante, factores como el avance en las tecnologías de computación visual y aprendizaje automatizado, el aumento en la capacidad de cómputo de los dispositivos móviles y la promoción de políticas de inclusión y accesibilidad en distintas organizaciones estatales e internacionales han influenciado a que cada vez se estudie más esta área de conocimiento.

El conjunto de datos utilizado para este estudio fue generado a partir de una combinación de recursos públicos en línea y de videos de autoría propia. La diversidad en los datos obtenidos y generados juega un rol importante en el resultado final del modelo de aprendizaje automatizado desarrollado, variaciones como la rotación del cuerpo en la toma, la luminosidad del entorno y variaciones en los movimientos de los gestos realizados ayudan a que el modelo de aprendizaje automatizado pueda generalizar de mejor forma los datos procesados y así tener una mejor predicción como resultado.

De acuerdo con el estudio realizado, las redes neuronales recurrentes como las redes neuronales de memoria a corto plazo extendido brindan un rendimiento superior en comparación con otros modelos como las redes neuronales convolucionales tradicionales. Esto es debido a que las redes neuronales recurrentes son más ligeras en términos de capacidad de cómputo, por lo que son ideales al momento de desarrollar sistemas de detección de gestos en lengua de signos en tiempo real.

5.1 Consecución de los objetivos planteados

Este estudio se enfocó en el desarrollo de un sistema de reconocimiento de gestos en lengua de signos española. Este sistema fue desarrollado utilizando técnicas inteligencia artificial, aprendizaje automático y computación visual para detectar, interpretar y traducir gestos de ciertas palabras en la lengua lenguaje de signos española en tiempo real. El proceso de detección comienza con la captura de los movimientos de las manos, torso y expresiones faciales utilizando la cámara web de un ordenador portátil. Estas entradas son procesadas por una red neuronal de arquitectura recurrente y densa que analizan las características espaciales y temporales de los gestos para finalmente. Finalmente, la predicción de este modelo de aprendizaje automático se muestra en pantalla para visualización del usuario.

Durante la fase de investigación de este trabajo, se realizó una búsqueda exhaustiva del estado del arte de los sistemas de reconocimiento de lengua de signos en general. Los sistemas actuales de este tipo utilizan algoritmos avanzados de aprendizaje automático; específicamente en el dominio del aprendizaje profundo, como es el caso de las redes neuronales convolucionales y redes neuronales recurrentes para procesar y comprender movimientos complejos de las manos, torso y expresiones faciales de los signantes. El

reconocimiento en tiempo real hoy en día se puede lograr mediante la integración de cámaras de vídeo, guantes con sensores de movimiento, sensores con rastreo de movimiento o cámaras estereoscópicas, lo que permite la traducción instantánea de un conjunto específico de gestos en lengua de signos a texto u otro formato.

Asimismo, durante este estudio se elaboró un conjunto de datos con gestos en lengua de signos española para el proceso de entrenamiento del modelo de aprendizaje automático seleccionado para el sistema propuesto. Estos sistemas se entrenan con conjuntos de datos extensos y anotados, que incluyen una amplia variedad de movimientos de las manos, brazos y expresiones faciales. El conjunto de datos elaborado consistió en una colección de siete expresiones en lengua de signos española, los gestos fueron tomados tanto de recursos en línea como de grabaciones propias mediante la cámara web de mi ordenador portátil. Para cada gesto del vocabulario se guardaron 30 vídeos y grabaciones de 30 fotogramas cada una, al procesar todos estos vídeos para obtener los valores espaciales de las posiciones de las manos, torso y rostro, se cuenta con un total de 10.470.600 valores en el conjunto de datos elaborado.

Finalmente, durante este estudio se exploraron distintas técnicas para la detección de gestos en lengua de signos en tiempo real. Las técnicas de aprendizaje automático e inteligencia artificial predominan en este aspecto, siendo las más utilizadas las redes neuronales convoluciones y las redes neuronales recurrentes para el procesamiento de los valores espaciales y temporales del conjunto de datos. Sin embargo, diseños específicos de este tipo de redes neuronales son de utilidad al momento de desarrollar un sistema que interprete lengua de signos en tiempo real, como es el caso de las redes neuronales de memoria a corto plazo extendida, ya que es una red neuronal relativamente ligera que permite obtener predicciones confiables en un tiempo menor al de otras redes neuronales utilizadas para este propósito.

5.2 Problemas encontrados

Si bien los objetivos de este trabajo de investigación fueron alcanzados de forma exitosa, durante el proceso de elaboración de este estudio se encontraron algunas limitantes para la detección de gestos en lengua de signos. En primer lugar, actualmente los vocabularios que existen en la lengua de signos española se limitan a palabras en concreto en lugar de oraciones con una estructura gramatical más compleja.

De igual forma, los recursos encontrados en internet son elaborados por signantes con un alto dominio de la lengua de signos española son grabados en un entorno ideal, dando un alto contraste entre el fondo y la vestimenta del signante. Esto genera una carencia en la variedad de los datos considerando condiciones del entorno como un signante con vestimenta de colores claros sobre un fondo blanco o bien variaciones en los gestos de los signantes con un dominio inferior en la lengua de signos española.

Asimismo, el sistema desarrollado actualmente no detecta cuando un signante deja de expresarse en cámara, por lo que si se tiene una posición del cuerpo en descanso el sistema intenta dar una predicción con la información que tiene en el momento, dando así una predicción de una expresión que no se está haciendo.

5.3 Conclusiones personales

Este trabajo de investigación surgió como una posible aplicación de las técnicas de inteligencia artificial y computación visual que había conocido durante mis estudios de grado, especialmente cuando ingresaba a mis redes sociales y

notaba a personas que conozco haciendo transmisiones en vivo conversando exclusivamente en lengua de signos.

Sistemas de reconocimiento de gestos en lengua de signos como los presentados en este estudio tienen el potencial de ser una herramienta robusta para aumentar la accesibilidad de los servicios que consumimos de uso diario, especialmente cuando se trata de la reproducción de contenido audiovisual en nuestros dispositivos móviles. Dada la naturaleza del funcionamiento de las redes neuronales, este sistema tiene la capacidad de ser adaptado a cualquier otra lengua de signos, como ser la Lengua de Señas Hondureña (LESHO) que es la lengua de signos de las personas sordas de mi país natal.

Por lo anterior, este trabajo representa el primer paso para la elaboración de un sistema que tiene el potencial de poder ser utilizado por todas las personas que no tenemos un dominio pleno de la lengua de signos utilizada en nuestras regiones respectivas. Este estudio muestra una prueba de concepto que puede ser aplicado en España, Honduras o cualquier otro país que desee explorar los beneficios en accesibilidad para contenido audiovisual.

5.4 Líneas futuras

Uno de los aspectos a futuro que pueden mejorar los resultados del sistema desarrollado en este estudio es la expansión del conjunto de datos utilizado para entrenar el modelo de aprendizaje automático. Por ejemplo, se puede aumentar la cantidad de gestos y expresiones del vocabulario con distintos niveles de complejidad gramatical en lugar de enfocarse en los siete gestos originales del conjunto de datos de este estudio.

Además de incluir más expresiones en el vocabulario, la inclusión de signantes que varíen en género, edad y dominio de la lengua de signos española ayudará a expandir las capacidades de la red neuronal cuando sea entrenada con este conjunto de datos, ofreciendo una mayor robustez en la detección de los gestos.

Otro aspecto a futuro es la exploración de las técnicas empleadas en este estudio para otras lenguas de signos que no tienen un conjunto de datos definido para este propósito como es el caso de Honduras. Actualmente no se cuenta con un repositorio que permita ver contenido audiovisual en LESHO ni un sistema que permita detectar gestos en esta lengua para facilitar la comprensión de las personas sordas hacia otros usuarios que no están familiarizados con la lengua de signos.

Asimismo, este tipo de sistemas ofrece la oportunidad de explorar la funcionalidad de detección de gestos en dispositivos móviles, ya que estos son los más utilizados por las personas en todo el mundo. Especialmente porque las cámaras de vídeo de los dispositivos móviles actuales son superiores a las encontradas en la mayoría de los ordenadores portátiles.

Finalmente, este sistema puede ser explorado para una posible implementación en la nube que permita la traducción en tiempo real utilizando servicios web, similares a los utilizados en sistemas de traducción en tiempo real de texto como Google Translate. Esto permitiría que el sistema reduzca drásticamente las especificaciones técnicas requeridas para ejecutar este tipo de sistemas, pudiendo ser utilizado en dispositivos móviles de gama baja.

6 Bibliografía

- Farooq, U., Mohd, M. S., Sabir, N., Hussain, A., & Abid, A. (2021). Advances in machine translation for sign language: approaches, limitations, and challenges. *Neural Computing and Applications*, 14357–14399.
- Adeyanju, I., Bello, O., & Adegboye, M. (2021). Machine learning methods for sign language recognition: A critical review and analysis. *Intelligent Systems with Applications*.
- Al-Hammadi, M., Muhammad, G., Abdul, W., Alsulaiman, M., Bencherif, M. A., & Mekhtiche, M. A. (2020). Hand Gesture Recognition for Sign Language Using 3DCNN. *IEEE Access*, 8, 79491-79509. doi:10.1109/ACCESS.2020.2990434
- Bansal, M., Kumar, M., & Kumar, M. (2021). 2D object recognition: a comparative analysis of SIFT, SURF and ORB feature descriptors. *Multimedia Tools and Applications*, 18839–18857.
- Bragg, D., Koller, O., Bellard, M., Berke, L., Boudreault, P., Braffort, A., . . . Morris, M. R. (2019). Sign Language Recognition, Generation, and Translation: An Interdisciplinary Perspective. *ASSETS*, 16-31.
- Brownlee, J. (20 de Julio de 2018). *What is the Difference Between a Batch and an Epoch in a Neural Network?* Obtenido de Deep Learning: https://deeplearning.lipingyang.org/wp-content/uploads/2018/07/What-is-the-Difference-Between-a-Batch-and-an-Epoch-in-a-Neural-Network_.pdf
- Burbank Unified School District. (s.f.). *Burbank Unified School District*. Obtenido de ASL Alphabet: https://www.burbankusd.org/cms/lib/CA50000426/Centricity/Domain/884/ASL_ALPHABET__John_Muir.pdf
- Centro de Normalización de Lingüística de la Lengua de Signos Española. (2022). *Informe de valoración jurídica de la Ley 27/2007, de 23 de octubre*. CNLSE.
- Centro de Normalización Lingüística de la Lengua de Signos Española. (2023). *Corpus de la Lengua de Signos Española (CORLSE): El desafío de documentar una lengua visual*. Madrid: CNLSE.
- Chassagnon, G., Vakalopoulou, M., Paragios, N., & Revel, M.-P. (2020). Deep learning: definition and perspectives for thoracic imaging. *European Radiology*, 2021–2030.
- Chen, J., Kellokumpu, K., Zhao, G., & Pietikäinen, M. (2013). *RLBP: Robust Local Binary Pattern*. Oulu: Center for Machine Vision Research.
- Confederación Estatal de Personas Sordas. (s.f.). *Historia*. Obtenido de Confederación Estatal de Personas Sordas: <https://www.cnse.es/index.php/cnse/historia>
- Confederación Estatal de Personas Sordas. (s.f.). *Lengua de Signos*. Obtenido de Confederación Estatal de Personas Sordas: <https://www.cnse.es/index.php/lengua-de-signos>
- Cortés, Y. M., & Barreto, A. G. (2013). Variación Sociolingüística En La Lengua De Señas Colombiana: Observaciones Sobre El Vocabulario Deportivo, En El Marco De La Planificación Lingüística. *Forma y Función*, 149-170.
- European Commission. (2019). *A Definition of AI: Main Capabilities and Scientific Disciplines*. Brussels: European Commission.

- European Commission. (2020). *AI Watch: Defining Artificial Intelligence*. Luxembourg: European Union.
- Fakhfakh, S., & Jemaa, Y. B. (2018). Gesture Recognition System for Isolated Word Sign Language. *Computación y Sistemas*, 1415–1430.
- Fradkov, A. (2020). Early History of Machine Learning. *IFAC PapersOnline*, 1385-1390.
- Fundación de la Conferación Estatal de Personas Sordas. (s.f.). *Fundación de la Conferación Estatal de Personas Sordas*. Obtenido de Diccionario de la Lengua de Signos Española: <https://fundacioncnse-dilse.org/>
- Gobierno de España. (24 de octubre de 2007). Ley 27/2007, de 23 de octubre, por la que se reconocen las lenguas de signos españolas y se regulan los medios de apoyo a la comunicación oral de las personas sordas, con discapacidad auditiva y sordociegas.
- González, M. Á. (2003). *Lenguaje de Signos*. Alicante: Biblioteca Virtual Miguel de Cervantes.
- Google. (s.f.). *MediaPipe Solutions Guide*. Obtenido de Google AI Edge: <https://ai.google.dev/edge/mediapipe/solutions/guide>
- Harish, M. N., & Poonguzhali, S. (2022). Review of gesture recognition gloves and its applications in thermal. *Materials Today: Proceedings*, 1002-1011.
- He, K., Zhang, X., Ren, S., & Sun, J. (2015). *Deep Residual Learning for Image Recognition*. Microsoft.
- Jeyasheeli, G., & Indumathi, N. (2021). Deep Learning Based Indian Sign Language Words Identification System. *Recent Trends in Intensive Computing*, 738-743.
- Kothadiya, D., Bhatt, C., Sapariya, K., Patel, K., Gil-González, A.-B., & Corchado, J. (2022). Deepsign: Sign Language Detection and Recognition Using Deep Learning. *Electronics*.
- Kumar, P., Gaubaa, H., Pratim, P., & Prosad, D. (2017). A multimodal framework for sensor based sign language recognition. *Neurocomputing*, 21-38.
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 436-444.
- Ministerio de Derechos Sociales y Agenda 2030. (2022). *Estrategia Española Sobre Discapacidad 2022-2030 Para el Acceso, Goce y Disfrute de los Derechos Humanos de las Personas con Discapacidad*. Gobierno de España.
- Miron, C., Păsărică, A., Arotăriței, D., Costin, H., Bozomitu, R., & Rotariu, C. (2017). Hand Gesture Detection using a Stereo Camera System and Simulation of Movement. *The 10th International Symposium On Advanced Topics In Electrical Engineering*, 297-300.
- Mohamed, N., Mustafa, M. B., & Jomhari, N. (2021). A Review of the Hand Gesture Recognition System: Current Progress and Future Directions. *IEEE Access*, 9, 157422-157436. doi:10.1109/ACCESS.2021.3129650
- Mohamed, N., Mustafa, M., & Jomhari, N. (2021). A Review of the Hand Gesture Recognition System: Current Progress and Future Directions. *IEEE Access*, 157422-157436.

- Mor, B., Garhwal, S., & Kumar, A. (2020). A Systematic Review of Hidden Markov Models and Their Applications. *Archives of Computational Methods in Engineering*, 1429–1448.
- Noraini, M., Mumtaz, M. B., & Jomhari, N. (2021). A Review of the Hand Gesture Recognition System: Current Progress and Future Directions. *IEEE Access*, 157422-157436.
- Núñez-Marcos, A., Perez-de-Viñaspre, O., & Labaka, G. (2023). A survey on Sign Language machine translation. *Expert Systems with Applications*.
- ONU. (2015). *Resolución aprobada por la Asamblea General el 25 de septiembre de 2015*. Organización de las Naciones Unidas. Obtenido de <https://documents.un.org/doc/undoc/gen/n15/291/93/pdf/n1529193.pdf?token=qcRfvltzjooKRXkBdT&fe=true>
- Organización Panamericana de la Salud. (2021). *Informe Mundial Sobre la Audición*. Washington, D.C.: Organización Panamericana de la Salud.
- Pan, T.-Y., Lo, L.-Y., Yeh, C.-W., Li, J.-W., Liu, H.-T., & Hu, M.-C. (2016). Real-time Sign Language Recognition in Complex Background Scene Based on a Hierarchical Clustering Classification Method. *IEEE Second International Conference on Multimedia Big Data*, 64-67.
- Park, H., Lee, Y., & Ko, J. (2021). Enabling Real-time Sign Language Translation on Mobile Platforms with On-board Depth Cameras. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*.
- Pisner, D. A., & Schnyer, D. M. (2020). Chapter 6 - Support vector machine. En D. A. Pisner, & D. M. Schnyer, *Machine Learning: Methods and Applications to Brain Disorders* (págs. Pages 101-121). Academic Press.
- Raghuveera, T., Deepthi, R., Mangalashri, R., & Akshaya, R. (2020). A depth-based Indian Sign Language recognition using Microsoft Kinect. *Sādhana*.
- Rana, R., Epps, J., Jurdak, R., Li, X., Goecke, R., Brereton, M., & Soar, J. (2016). Gated Recurrent Unit (GRU) for Emotion Classification from Noisy Speech. *ArXiv*.
- Rastgoo, R., Kiani, K., & Escalera, S. (2021). Sign Language Recognition: A Deep Survey. *Expert Systems With Applications*.
- Renotte, N. (19 de Junio de 2021). *Action Detection for Sign Language*. Obtenido de <https://github.com/nicknochnack/ActionDetectionforSignLanguage>
- Rodríguez, I., & Roche, J. M. (2007). El uso educativo de la Lengua de Signos Española (LSE) y su problemática. *Revista de Educación*, 419-441.
- Rodríguez, M. Á. (1992). *Lenguaje de signos*.
- Saeed, Z., Zainol, Z., Zaidan, B., & Alamoodi, A. (2022). A Systematic Review on Systems-Based Sensory Gloves for Sign Language Pattern Recognition: An Update From 2017 to 2022. *IEEE Access*, 123358-123377.
- Samaan, G., Wadie, A., Attia, A., Asaad, A., Kamel, A., Slim, S., . . . Cho, Y.-I. (2022). MediaPipe's Landmarks with RNN for Dynamic Sign Language Recognition. *Electronics*.
- Siarni-Namini, S., Tavakoli, N., & Namin, A. (2019). The Performance of LSTM and BiLSTM in Forecasting Time Series. *IEEE International Conference on Big Data*, 3285-3292.

- Valdés-González, A., & Martín-Antón, J. (2020). Lengua de Signos Española y ámbitos específicos. Una propuesta multidisciplinar e inclusiva para la búsqueda, análisis y creación de Signos. *Aula Abierta*, 159-170.
- Vercher, E. (11 de Abril de 2018). *La Lengua de Señas*. Obtenido de <https://www.agestrad.com/la-lengua-de-senas/>
- Wang, H., Jia, S., & Wang, Z. (2016). Multi-sensor Formation Targets Template Matching Tracking Algorithm. *International Conference on Mechatronics, Control and Automation Engineering*, 141-146.
- Wang, S., Wang, K., Yang, T., Li, Y., & Fan, D. (2022). Improved 3D-ResNet sign language recognition algorithm with enhanced hand features. *Scientific Reports*.
- Yadav, J., & Sharma, M. (2013). A Review of K-mean Algorithm. *International Journal of Engineering Trends and Technology* , 2972-2976.
- Yin, S., Yang, J., Qu, Y., Wenjun, L., Yunfei, G., Liu, H., & Wei, D. (2018). Research on Gesture Recognition Technology of Data Glove Based on Joint Algorithm. *Advances in Engineering Research*, 13-22.
- Zhang, L., Zhou, W., Li, J., Li, J., & Lou, X. (2020). Histogram of Oriented Gradients Feature Extraction Without Normalization. *IEEE Asia-Pacific Conference on Circuits and Systems*, 252-255.

7 Anexos

7.1 Código del sistema desarrollado para la detección de gestos en lengua de signos española

1. Importación de librerías

Nota: En caso de necesitar instalar las librerías se puede ejecutar el siguiente bloque antes de realizar las importaciones:

```
!pip install opencv-python==4.10.0.82 numpy==1.23.5 mediapipe==0.10.14 tensorflow==2.12.0rc0
```

```
import os
import cv2
import time
import math
import numpy as np
import mediapipe as mp
import tensorflow as tf
from datetime import datetime
from matplotlib import pyplot as plt

# Para la partición de datos de entrenamiento y prueba
from sklearn.model_selection import train_test_split
from tensorflow.keras.utils import to_categorical

# Para la creación del modelo con LSTM
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM, Dense
from tensorflow.keras.callbacks import TensorBoard

# Para la evaluación del modelo con LSTM
from sklearn.metrics import multilabel_confusion_matrix, accuracy_score
```

2. Uso de OpenCV para detectar partes del cuerpo

2.1. Detección de puntos de interés (landmarks) de cara, brazos y manos con OpenCV

```
def body_detection(image, model):
    image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB) # Se convierte porque CV2 trabaja con BGR y el modelo con RGB
    image.flags.writeable = False
    results = model.process(image)
    image.flags.writeable = True
    image = cv2.cvtColor(image, cv2.COLOR_RGB2BGR) # Se regresa al valor por defecto BGR de CV2

    return image, results
```

```
def draw_landmarks(image, results):
    mp_draw.draw_landmarks(
        image,
        results.face_landmarks,
        mp_hol.FACEMESH_TESSELATION,
        mp_draw.DrawingSpec(color=(255,255,255), thickness=1, circle_radius=1),
        mp_draw.DrawingSpec(color=(255,255,255), thickness=1, circle_radius=1)
    )
    mp_draw.draw_landmarks(
        image,
        results.pose_landmarks,
        mp_hol.POSE_CONNECTIONS,
        mp_draw.DrawingSpec(color=(255,0,0), thickness=1, circle_radius=1),
        mp_draw.DrawingSpec(color=(255,0,0), thickness=1, circle_radius=1)
    )
    mp_draw.draw_landmarks(
        image,
        results.left_hand_landmarks,
        mp_hol.HAND_CONNECTIONS,
        mp_draw.DrawingSpec(color=(0,255,0), thickness=1, circle_radius=1),
        mp_draw.DrawingSpec(color=(0,255,0), thickness=1, circle_radius=1)
    )
    mp_draw.draw_landmarks(
        image,
        results.right_hand_landmarks,
        mp_hol.HAND_CONNECTIONS,
        mp_draw.DrawingSpec(color=(0,0,255), thickness=1, circle_radius=1),
        mp_draw.DrawingSpec(color=(0,0,255), thickness=1, circle_radius=1)
    )
```

2.2. Extracción de puntos de interés (landmarks) a un arreglo de numpy para ser utilizado por la LSTM neural network

```
def get_landmarks_array(results):
    face = np.array([[r.x, r.y, r.z] for r in results.face_landmarks.landmark]).flatten() if results.face_landmarks else np.zeros(468*3)
    pose = np.array([[r.x, r.y, r.z, r.visibility] for r in results.pose_landmarks.landmark]).flatten() if results.pose_landmarks else np.zeros(33*4)
    left_hand = np.array([[r.x, r.y, r.z] for r in results.left_hand_landmarks.landmark]).flatten() if results.left_hand_landmarks else np.zeros(21*3)
    right_hand = np.array([[r.x, r.y, r.z] for r in results.right_hand_landmarks.landmark]).flatten() if results.right_hand_landmarks else np.zeros(21*3)
    return np.concatenate([face, pose, left_hand, right_hand])
```

```
mp_hol = mp.solutions.holistic
mp_draw = mp.solutions.drawing_utils
holistic = mp_hol.Holistic(min_detection_confidence=0.7, min_tracking_confidence=0.7)
```

Nota: Este bloque de código es para verificar que los landmarks se están mostrando correctamente en la pantalla, se puede omitir en la ejecución del programa

```
cap = cv2.VideoCapture(0)

width = cap.get(cv2.CAP_PROP_FRAME_WIDTH) # float 'width'
height = cap.get(cv2.CAP_PROP_FRAME_HEIGHT) # float 'height'

while cap.isOpened():
    # Obtiene la imagen de la camara
    ret, frame = cap.read()

    # Realiza la detección de rostro, brazos y manos con mediapipe
    image, results = body_detection(frame, holistic)

    # Dibuja los puntos detectados con el modelo holístico y los agrega a la captura de video
    draw_landmarks(image, results)
```

```

cv2.putText(image, f'Size {width}x{height}', (15,12), cv2.FONT_HERSHEY_SIMPLEX, 0.5, (255,255,255), 1, cv2.LINE_AA)

# Muestra captura de video en pantalla
cv2.imshow('OpenCV', image)

# En caso de oprimir ESC, sale del while
if cv2.waitKey(27) & 0xFF == ord('\x1b'):
    break

cap.release()
cv2.destroyAllWindows()

```

3. Generación de conjunto de datos para el entrenamiento y validación

Nota: las secciones 3.2. a la sección 3.5. pueden omitirse si ya se generaron datos de entrenamiento

3.1. Configuración de la carpeta de datos

```

# Variables de open CV y MediaPipe para obtener Los Landmarks
cap = cv2.VideoCapture(0)
mp_hol = mp.solutions.holistic
mp_draw = mp.solutions.drawing_utils
holistic = mp_hol.Holistic(min_detection_confidence=0.7, min_tracking_confidence=0.7)

# Ruta de la carpeta de datos
folder_path = os.path.join('data')

# Configuración del numero de secuencias a generar por cada gesto y la cantidad de frames a recolectar por cada secuencia
gestures = np.array(os.listdir(folder_path))
seq_amount = 30 # 30 secuencias por gesto
seq_length = 30 # 30 frames por secuencia

```

3.2. Creación de carpetas

```

for g in gestures:
    for s in range(seq_amount):
        try:
            os.makedirs(os.path.join(folder_path, g, str(s)))
        except:
            pass

```

3.3. Programa para la captación de datos desde la cámara web

```

for g in gestures:
    for s in range(seq_amount):
        for frame_number in range(seq_length):

            # Obtiene la imagen de la camara
            ret, frame = cap.read()

            # Realiza la detección de rostro, brazos y manos con mediapipe
            image, results = body_detection(frame, holistic)

            # Dibuja los puntos detectados con el modelo holístico y los agrega a la captura de video
            draw_landmarks(image, results)

            # Inicia la recolección de datos de los gestos, muestra en pantalla mensaje para hacer gestos y espera 1 segundo para tomar nuevos datos
            if frame_number == 0:
                cv2.putText(image, 'Starting data gathering', (120,200), cv2.FONT_HERSHEY_SIMPLEX, 1, (255,255,255), 4, cv2.LINE_AA)
                cv2.putText(image, f'Gathering frames for {g} - Video No. {s}', (15,12), cv2.FONT_HERSHEY_SIMPLEX, 0.5, (255,255,255), 1, cv2.LINE_AA)
                cv2.waitKey(1000)
            else:
                cv2.putText(image, f'Gathering frames for {g} - Video No. {s}', (15,12), cv2.FONT_HERSHEY_SIMPLEX, 0.5, (255,255,255), 1, cv2.LINE_AA)

            # Guarda los datos obtenidos en un archivo
            keypoints = get_landmarks_array(results)
            output_path = os.path.join(folder_path, g, str(s), str(frame_number))
            np.save(output_path, keypoints)

            # Muestra captura de video en pantalla
            cv2.imshow('OpenCV', image)

            # En caso de oprimir ESC, sale del while
            if cv2.waitKey(27) & 0xFF == ord('\x1b'):
                break

cap.release()
cv2.destroyAllWindows()

```

3.4. Programa para la captación de datos desde la carpeta de videos

```

videos_path = os.path.join('videos')
output_videos_path = os.path.join('data_videos')

for gesture in os.listdir(videos_path):
    for video in os.listdir(os.path.join(videos_path, gesture)):
        cap = cv2.VideoCapture(os.path.join(videos_path, gesture, video))

        print(f'Video: {gesture}\nTotal de frames: {cap.get(cv2.CAP_PROP_FRAME_COUNT)}\nTomar un frame cada {int(cap.get(cv2.CAP_PROP_FRAME_COUNT))/seq_length} frames')

        while cap.isOpened():
            ret, frame = cap.read()

            if not ret:
                print("Can't receive frame (stream end?). Exiting ...")
                break

            # Cambia la resolución a 640x480 px
            current_frame = int(cap.get(cv2.CAP_PROP_POS_FRAMES))
            frame = cv2.resize(frame, (640,480))

            # Realiza la detección de rostro, brazos y manos con mediapipe
            image, results = body_detection(frame, holistic)

```

```

# Dibuja los puntos detectados con el modelo holístico y los agrega a la captura de video
draw_landmarks(image, results)

cv2.putText(image, f'Frame {current_frame} of {cap.get(cv2.CAP_PROP_FRAME_COUNT)}', (15,12), cv2.FONT_HERSHEY_SIMPLEX, 0.5, (255,255,255), 1, cv2.LINE_AA)

# Como cada secuencia tiene 30 frames, se recortan algunos frames de forma dinámica
if (current_frame-1) % math.floor((cap.get(cv2.CAP_PROP_FRAME_COUNT)/seq_length)) == 0:
    # Guarda los keypoints en archivos
    keypoints_videos = get_landmarks_array(results)
    output_path = os.path.join(output_videos_path, gesture, video.split('.')[0])
    if not os.path.exists(output_path):
        try:
            os.makedirs(output_path)
        except:
            pass
    np.save(os.path.join(output_path, str(current_frame-1)), keypoints_videos)

    cv2.imshow('frame', image)

    if cv2.waitKey(1) == ord('q'):
        break

cap.release()
cv2.destroyAllWindows()

```

3.5. Eliminación de frames sobrantes por cada video, a modo de tener un número fijo de frames

```

def get_id(filename):
    return int(filename.split('.')[0])

# Recorre cada una de las carpetas para eliminar los primeros N y últimos M frames guardados, a modo de sólo dejar 30 frames por video
for gesture in os.listdir(output_videos_path):
    for video in os.listdir(os.path.join(output_videos_path, gesture)):
        files = sorted(os.listdir(os.path.join(output_videos_path, gesture, video)), key=get_id)
        n_files_delete = len(files)-seq_length

        if n_files_delete > 0:
            print(f'Deleting files...')
            if n_files_delete % 2 == 0:
                print(f'Deleting {n_files_delete} - {int(n_files_delete/2)} at the beginig and {math.ceil(n_files_delete/2)} at the end')
                files_to_delete = files[:int(n_files_delete/2)] + files[-int(n_files_delete/2):]
                print('Deleting files', files_to_delete)
                for f in files_to_delete:
                    os.remove(os.path.join(output_videos_path, gesture, video, f))
            else:
                print(f'Deleting {n_files_delete} - {math.floor(n_files_delete/2)} at the beginig and {math.ceil(n_files_delete/2)} at the end')
                files_to_delete = files[:math.floor(n_files_delete/2)] + files[-math.ceil(n_files_delete/2):]
                print('Deleting files', files_to_delete)
                for f in files_to_delete:
                    os.remove(os.path.join(output_videos_path, gesture, video, f))
        else:
            print('Already 30 files per sequence')

        print('Renaming files...')
        files = sorted(os.listdir(os.path.join(output_videos_path, gesture, video)), key=get_id)
        name_seq = 0
        for index, file in enumerate(files):
            print(f'Renaming {file} to {index}.npv')
            os.rename(os.path.join(output_videos_path, gesture, video, file), os.path.join(output_videos_path, gesture, video, f'{index}.npv'))

```

4. Preprocesamiento y preparación de datos

4.1. Obtención de los datos generados en la sección 3. Generación de conjunto de datos para el entrenamiento y validación

```

label_map = {label:num for num, label in enumerate(gestures)}
sequences = []
labels = []

for gesture in gestures:
    for sequence in range(seq_amount):
        window = []
        for frame_number in range(seq_length):
            res = np.load(os.path.join(folder_path, gesture, str(sequence), f'{frame_number}.npv'))
            window.append(res)
        sequences.append(window)
        labels.append(label_map[gesture])

```

4.2. Partición del conjunto de datos en datos de entrenamiento y datos de prueba

```

x = np.array(sequences)
y = to_categorical(labels).astype(int)
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.1)

```

5. Construcción del modelo con LSTM para la predicción de gestos en lengua de signos

5.1. Definición de los parámetros para guardar los logs en TensorBoard

```

logs_path = os.path.join('logs')
tb_callback = TensorBoard(log_dir=logs_path)

```

5.2. Configuración del modelo con LSTM

```

model = Sequential()
model.add(LSTM(64, return_sequences=True, activation='relu', input_shape=(30,1662)))
model.add(LSTM(128, return_sequences=True, activation='relu'))
model.add(LSTM(64, return_sequences=False, activation='relu'))
model.add(Dense(64, activation='relu'))

```

```
model.add(Dense(32, activation='relu'))
model.add(Dense(gestures.shape[0], activation='softmax'))
model.compile(optimizer='Adam', loss='categorical_crossentropy', metrics=['categorical_accuracy'])
```

5.3. Entrenamiento del modelo de predicción de gestos en lengua de signos

Nota: En caso de tener un modelo ya entrenado, se puede omitir esta parte y pasar a la sección 5.4.

```
model.fit(x_train, y_train, epochs=500, callbacks=[tb_callback])
filename = 'predictor_LSE_'+datetime.now().strftime('%d-%m-%Y_%H%M')+'.h5'
model.save(filename)
```

5.4. Carga del modelo entrenado

Nota: Ejecutar este código en caso de ya contar con un modelo entrenado

```
model.load_weights('./predictor_LSE_28-06-2024_0456.h5')
```

5.5. Evaluación del modelo entrenado

```
pred_test = model.predict(x_test)
pred_test_real = np.argmax(y_test, axis=1).tolist()
pred_test = np.argmax(pred_test, axis=1).tolist()
print("Matriz de confusión: \n", multilabel_confusion_matrix(pred_test_real, pred_test))
print("\n Precisión: ", accuracy_score(pred_test_real, pred_test))
```

6. Pruebas de reconocimiento en tiempo real

```
def render_prediction(res, gestures, frame):
    colors = [(16,37,64),
              (21,49,84),
              (26,61,105),
              (31,73,125),
              (36,85,145),
              (41,97,166),
              (46,109,186)]
    output = frame.copy()
    for n, p in enumerate(res):
        cv2.rectangle(output, (0,60+n*40), (int(p*100), 90+n*40), colors[n], -1)
        cv2.putText(output, gestures[n], (0,85+n*40), cv2.FONT_HERSHEY_SIMPLEX, 1, (255,255,255), 2, cv2.LINE_AA)
    return output
```

```
sequence = []
sentence = []
predictions = []
threshold = 0.999999
cap = cv2.VideoCapture(0)
```

```
while cap.isOpened():
    # Obtiene la imagen de la camara
    ret, frame = cap.read()

    # Realiza la detección de rostro, brazos y manos con mediapipe
    image, results = body_detection(frame, holistic)

    # Dibuja los puntos detectados con el modelo holístico y los agrega a la captura de video
    draw_landmarks(image, results)

    # Procesamiento de predicción
    landmarks = get_landmarks_array(results)
    sequence.append(landmarks)
    sequence = sequence[-30:]
    pred = model.predict(np.expand_dims(sequence, axis=0))[0]
    predictions.append(np.argmax(pred))

    if len(sequence) == 30:
        pred = model.predict(np.expand_dims(sequence, axis=0))[0]
        print(f'Prediction = {gestures[np.argmax(pred)]}, {pred}')
        predictions.append(np.argmax(pred))

    # Visualización de resultados
    if np.unique(predictions[-10:])[0] == np.argmax(pred):
        if pred[np.argmax(pred)] > threshold:
            if len(sentence) > 0:
                if gestures[np.argmax(pred)] != sentence[-1]:
                    sentence.append(gestures[np.argmax(pred)])
            else:
                sentence.append(gestures[np.argmax(pred)])

    if len(sentence) > 5:
        sentence = sentence[-5:]

    output_text = ' '.join(sentence)
    image = render_prediction(pred, gestures, image)

    # Muestra captura de video en pantalla
    cv2.imshow('OpenCV', image)

    # En caso de oprimir ESC, sale del while
    if cv2.waitKey(27) & 0xFF == ord('\x1b'):
        break

cap.release()
cv2.destroyAllWindows()
```