



UNIVERSIDAD AUTÓNOMA DEL
ESTADO DE MORELOS

UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS



Facultad reconocida
por sus programas de calidad

Nivel 1 de CIEES

FACULTAD DE CONTADURÍA, ADMINISTRACIÓN E INFORMÁTICA



UADY
UNIVERSIDAD
AUTÓNOMA
DE YUCATÁN

EN COLABORACIÓN CON:
FACULTAD DE CIENCIAS QUÍMICAS E INGENIERÍA, INSTITUTO NACIONAL DE
ASTROFÍSICA, ÓPTICA Y ELECTRÓNICA (INAOE) Y UNIVERSIDAD AUTÓNOMA
DE YUCATÁN (UADY)

TESIS:

“Modelos Computacionales Aplicados a la Lengua de Señas Mexicana”

DIRECTORES DE TESIS

DR. OUTMANE OUBRAM, Y DR. NOUREDDINE LAKOUARI

ALUMNO:

ING. MARIO EMMANUEL RODRÍGUEZ TREJO

POSGRADO:

MAESTRÍA EN OPTIMIZACIÓN Y CÓMPUTO APLICADO

Índice

ABSTRACTO.....	3
AGRADECIMIENTOS.....	4
INTRODUCCIÓN	5
<i>Antecedentes</i>	5
<i>Justificación</i>	5
<i>Problemática</i>	5
<i>Objetivos</i>	5
<i>Hipótesis</i>	6
<i>Alcance y limitaciones</i>	6
REVISIÓN LITERARIA.....	7
<i>Proceso general de reconocimiento de lengua de señas</i>	7
<i>Algoritmos de Aprendizaje de Máquina</i>	14
<i>Algoritmos de Aprendizaje Profundo</i>	16
<i>MediaPipe Holistic</i>	17
<i>Reconocimiento de lengua de señas con MediaPipe y LSM</i>	18
<i>Otros enfoques para la lengua de señas</i>	19
<i>Algoritmos del Estado del Arte con MediaPipe</i>	27
METODOLOGÍA.....	33
<i>Metodología de investigación</i>	33
<i>Propuesta de metodología de experimentación</i>	34
<i>Adquisición de datos</i>	34
<i>Preparación de datos</i>	35
<i>Extracción de características</i>	35
<i>Modelos de aprendizaje para la clasificación</i>	36
MÉTRICAS DE EVALUACIÓN	36
EXPERIMENTACIÓN Y RESULTADOS.....	37
<i>Bases de datos</i>	37
<i>Detalles de implementación</i>	38
<i>Configuración de los modelos</i>	41
<i>Análisis cuantitativo</i>	43
<i>Análisis comparativo</i>	45
DISCUSIÓN.....	45
CONCLUSIONES Y LIMITACIONES.....	47
<i>Resumen de contribuciones</i>	48
REFERENCIAS.....	49
APÉNDICE.....	57
<i>Apéndice 1. Arquitectura de Modelos LSTM y GRU</i>	57
<i>Apéndice 2. Resultados de las matrices de confusión</i>	58
LISTA DE FIGURAS	62
LISTA DE TABLAS.....	63

Abstracto

En México, alrededor de 2.4 millones de personas (1.9% de la población nacional) son sordas, y el apoyo de la Lengua de Señas Mexicana (LSM) es fundamental para las personas con discapacidades de comunicación. Se han desarrollado prototipos tecnológicos y de investigación de reconocimiento de lengua de señas para apoyar los sistemas de comunicación pública sin intérpretes humanos. Sin embargo, la mayoría de estos sistemas e investigaciones están estrechamente relacionados con la lengua de señas americana (LSA) u otras lenguas de señas de otros idiomas cuyo alcance ha tenido el más alto nivel de exactitud y reconocimiento de letras y palabras. El objetivo de la investigación actual es desarrollar y evaluar un sistema de reconocimiento de lengua de la dactilología y los primeros diez dígitos numéricos (1-10) de la LSM. Se creó una base de datos de lengua de signos y numeración de la LSM con los 29 caracteres diferentes de la dactilología de la LSM y los primeros diez dígitos con una cámara. Luego, *MediaPipe* se aplicó por primera vez para la extracción de funciones para ambas manos (21 puntos por mano). Una vez extraídas las características, se aplicaron técnicas de aprendizaje automático y aprendizaje profundo para reconocer las señales de LSM. Como resultado, el reconocimiento de patrones LSM en el contexto de señales estáticas (29 clases) y continuas (10 clases) arrojó una exactitud del 92 % con *Support Vector Machine* (SVM) y del 86 % con *Gated Recurrent Unit* (GRU). Los algoritmos entrenados se basan en escenarios completos con ambas manos; por lo tanto funcionarán bajo estas condiciones. Para mejorar la exactitud, se sugiere ampliar el número de muestras.

Agradecimientos

El autor desea agradecer al Consejo Nacional de Humanidades, Ciencias y Tecnologías de México (CONAHCyT) por financiar sus estudios con una beca de cobertura completa.

Adicionalmente, los supervisores y el autor queremos agradecer a la Unidad para la Inclusión Educativa y Atención a la Diversidad de la UAEM (Universidad Autónoma del Estado de Morelos), la Facultad de Comunicación Humana de la UAEM, y la Facultad de Matemáticas de la UADY (Universidad Autónoma de Yucatán) por su apoyo.

Agradecemos al Dr. Eliseo Guajardo Ramo, Director de la Unidad para la Inclusión Educativa y Atención a la Diversidad de la UAEM, a la Dra. Alma Janeth Moreno Aguirre, profesora investigadora de tiempo completo de la Facultad de Comunicación Humana de la UAEM, la Dra. Viridiana Aydeé León Hernández, Directora de la Facultad de Ciencias Químicas e Ingeniería (FCQeI) de la UAEM, y la Dra. María del Carmen Torres Salazar, Secretaría de Investigación de la FCQeI de la UAEM por su invaluable guía y apoyo a lo largo de este estudio.

Asimismo, agradecemos a las siguientes personas por su disponibilidad para generar muestras y realizar la anotación de datos y su facilitación de realizar la recopilación de muestras: Lic. Patricia Salazar Díaz (Licenciada en Pedagogía), Lic. María Fernanda Gutiérrez Álvarez (Licenciada en Comunicación Humana), M.S.Ed. Luz Patricia Capistrán Pérez (Maestría en Educación Especial), Kevin Manuel Rivera Ramirez, Angel Sanyey Diaz Ceballos, Ariana Rodriguez Rojas, Irasema Rodriguez Rojas, Maria Guadalupe Sarai Guillen Garcia, Mayra Vianey Huerta, Moses Martinez Martinez, Noreli Rodriguez Rojas , Ahinoam Concha Olascuaga, Roberto Carlos Valle Reyes, Peralta Miraselva Fernando y Jaimes Cortez Cristian Giovanni por sus aportes al proyecto de investigación.

Introducción

Antecedentes

La Lengua de Señas Mexicana (LSM) tiene antecedentes en la cultura Mesoamericana y Maya, donde se usaban las manos para la explicación de las fases de la luna (llena, menguante, etc).

Las raíces del desarrollo del LSM provienen de la lengua de señas francesa (LSF), iniciado desde 1800 con Abbe Charles L'Epee. En 1866, Don Eduardo Huet y Alcatraz migran a México estableciendo la primera escuela de sordos enseñando a tres estudiantes sordos, desarrollando así el LSM.

La importancia del LSM ha sido tan vital en México y en su historia que, en 2005, la Lengua de Señas Mexicana fue reconocida oficialmente como lengua nacional y es patrimonio lingüístico de México (COPRED, 2019).

Justificación

Acorde a la Organización Mundial de la Salud (WHO) 5% de la población mundial sufre de sordera, equivalente 450 millones de habitantes en el mundo (WHO, 2021). Además, se estima que en el 2050 habrá más de 700 millones de habitantes con sordera.

Enfocándonos en México, acorde al INEGI en el 2020, 2.4 millones de personas (1.9% aproximadamente de la población nacional) sufren de sordera (Mariano, M., 2020).

Dado lo anterior, se han formulado varias leyes, como la Ley General de Discapacidad del 2008, fracción XIII del Capítulo III, Artículo 10 de la última revisión a lo concerniente a la educación, se especifica que se debe “Impulsar programas de investigación, preservación y desarrollo de la lengua de señas, de las personas con discapacidad auditiva y de las formas de comunicación de las personas con discapacidad visual.”

Por tanto, desarrollar la presente investigación contribuye en al menos cinco de los diecisiete objetivos de Desarrollo Sostenible de la Agenda 2030 de Organización de las Naciones Unidas: 1) es el objetivo #4: Educación de calidad, 2) es el objetivo #8: Trabajo decente y crecimiento económico, 3) es el objetivo #9: Industria, innovación, e infraestructura, 4) es el objetivo #10: Reducción de las desigualdades, y 5) es el objetivo #11: Ciudades y comunidades sostenibles.

Problemática

En los últimos años, la Inteligencia Artificial (IA) ha evolucionado tanto en el desarrollo de los algoritmos de la visión computacional como en el Internet de las Cosas para la identificación de objetos y sus patrones. Sin embargo, en los últimos 4 años, las aplicaciones de los algoritmos no han tenido desarrollo de investigación en el estado del arte relevante en la Lengua de Señas Mexicana (LSM), pero sí un gran avance en la Lengua de Señas Americana (LSA) y otros, por lo que hay una vertiente en la aplicación eficiente de los algoritmos novedosos y paqueterías.

Objetivos

La vitalidad de la Lengua de Señas Mexicana en México y su unión con la tecnología para escalar la comunicación con los no hablantes de la lengua es propia para realizar un proyecto en cuestión. Por ello, se propone realizar el siguiente objetivo del proyecto de la maestría.

Objetivo General

- Implementar un sistema de interpretación de la dactilología (29 letras) y los primeros 10 dígitos (del 1 al 10) del LSM a través de visión computacional y con base en el estado del arte aplicado en otros LS que han probado ser los más eficientes en la exactitud de su reconocimiento con equipo de bajo coste computacional y de equipos.

Objetivos específicos

- Crear una base de datos de los 29 caracteres de la dactilología y los primeros 10 dígitos del LSM de foto y video para su posterior procesamiento, segmentación y modelaje.
- Experimentar con al menos 3 modelos del estado del arte de la lengua de señas que se puedan replicar para los caracteres y dígitos del LSM.
- Generar una tabla indicadores estadísticos para determinar el mejor modelo aplicado en la investigación acorde a los hiper parámetros y arquitecturas de modelos empleados.

Hipótesis

Dado los últimos estudios brindados en el LSA, la re-aplicación de sus algoritmos en el LSM podría generar un valor alto de exactitud del reconocimiento.

Alcance y limitaciones

Se tiene contemplado los 29 caracteres de la dactilología junto con los números del 1 al 10 para así lograr transmitir un mensaje dictado hacia otra persona. Donde, 29 son señas estáticas y 10 dinámicas (J, K, LL, Ñ, Q, RR, X, Z, 9 y 10). (Serafín , M. E., & González, R, 2011)

Revisión Literaria

La tecnología ha demostrado avances en el apoyo a la comunicación con hablantes de la LS mediante la construcción de sistemas que reconocen LS. En la literatura sistemática (Ahmed et al., 2018), se encontraron tres tipos de sistemas de reconocimiento de lengua de señas: guantes, visión artificial y sistemas híbridos. Los guantes han demostrado gran exactitud (+90%) y mejor precisión con sistemas híbridos (+99%). No obstante, la mayoría de las personas sordas no pueden permitirse guantes prácticos debido al precio, ya que solo se pueden permitir los gastos necesarios debido al nivel económico general de los hablantes de LS (Kim et al., 2018). Un ejemplo de los guantes es *Bright Sign Glove* (BrightSign Technology Limited, 23 de febrero de 2023), que cuesta más de USD 2,400.00 en línea.

En cuanto a la visión por computadora, sensores complejos como *Xbox One Kinect* y *Leap Motion Controller* han participado en muchos estudios para el reconocimiento de LS (Guzsvinecz et al., 2019). Sin embargo, el costo de adquisición para los usuarios finales aún no es asequible y escalable debido a los costos finales de más de USD 99,99 (UltraLeap, 23 de febrero de 2023; Xbox One, 23 de febrero de 2023).

Una revisión sistemática de los sistemas de reconocimiento de lengua de señas (Adeyanju et al., 2021) mencionó que la cámara web es el único dispositivo que ha demostrado bajo costo, excelente escalabilidad y más del 50% de participación en los artículos en su revisión sistemática de literatura. Una cámara web HD puede costar alrededor de USD 29,00 (EMEET, 23 de febrero de 2023); otras instituciones y personas pueden obtener una cámara integrada en sus teléfonos celulares o computadoras por defecto. Por lo tanto, se puede implementar un Sistema de reconocimiento de LS descargando solo el software, el repositorio de códigos o la aplicación.

Proceso general de reconocimiento de lengua de señas

Según Adeyanju et al. (2021), los pasos generales para un sistema de reconocimiento de lengua de signos son los siguientes:

- **Adquisición de datos:** Entradas de imágenes o videos y su almacenamiento.
- **Preprocesamiento:** Imágenes estandarizadas, filtradas o recortadas.
- **Segmentación:** Eliminación del fondo u otras partes.
- **Extracción de características:** Extracción de las características para su clasificación.
- **Clasificación:** Clasificación con un algoritmo acorde a la categoría predicha.

En adición a la anterior, se pudo observar otra en paralelo de Adeyanju et al. (2021). Donde se mencionan métodos complementarios y en adhesión la proveniencia y tendencia en la temática del reconocimiento de lengua de señas como se puede ver la siguiente Figura 1.

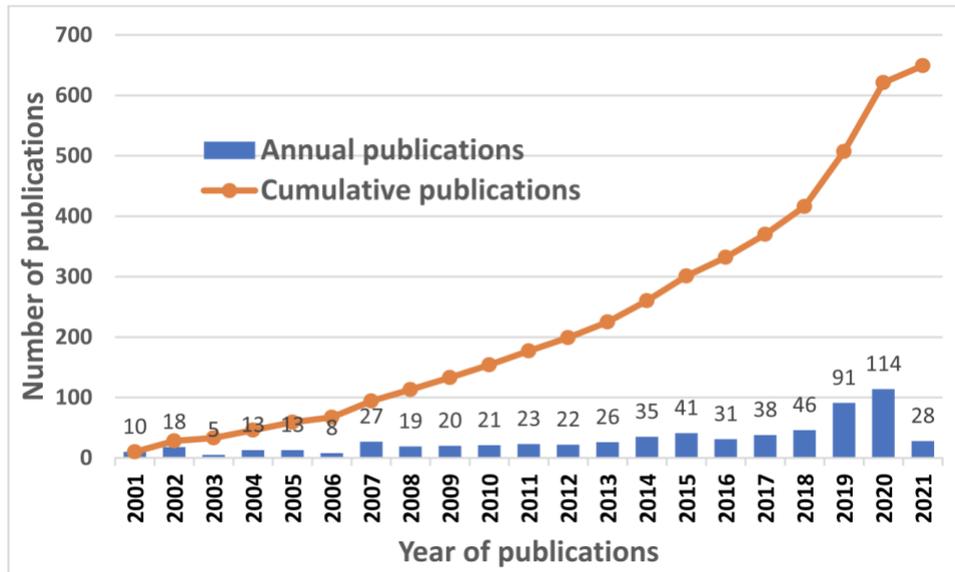


Figura 1. Número de publicaciones de artículos en relación a la lengua de señas de Adeyanju et al. (2021).

A continuación, en la Figura 2 se puede observar que entre los primeros países de contribuciones en reconocimiento de Lengua de Señas Mexicana no predomina en la investigación, pero India, China y USA lideran.

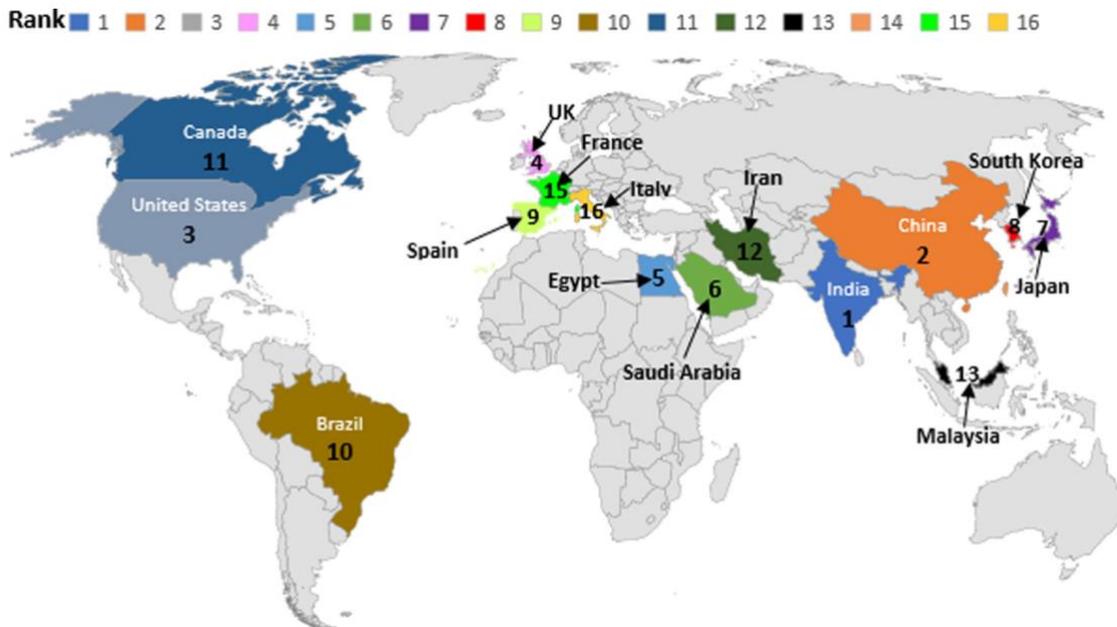


Figura 2. Países con mayor número de publicaciones de artículos en relación a la lengua de señas de Adeyanju et al. (2021).

En la siguiente Figura 3 se muestra la red de colaboración entre países en la temática de reconocimiento de lengua de señas, donde entre más cercano este al centro y mayor número de conexiones tenga, significa que entonces tiene mayor número de colaboraciones.

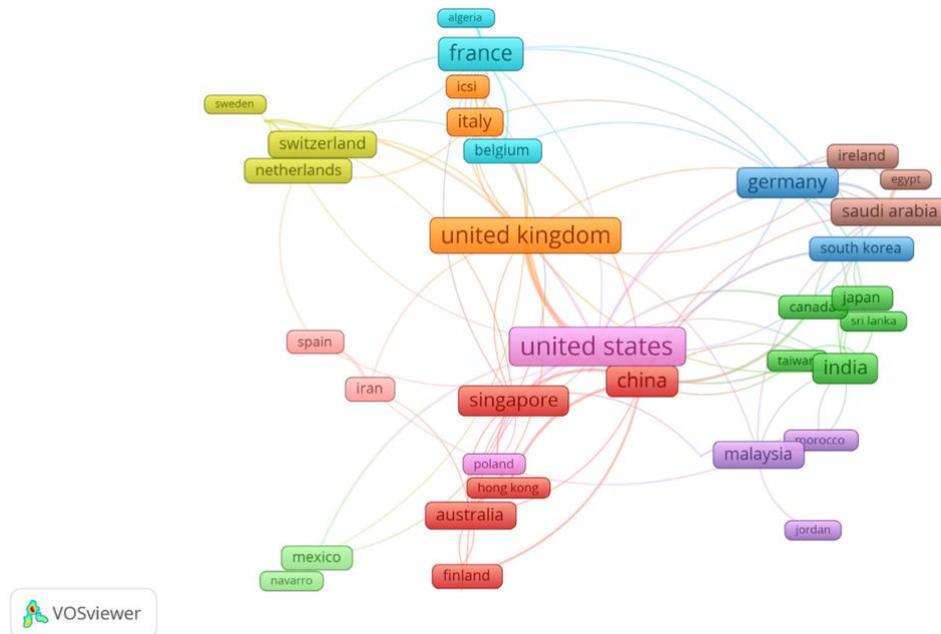


Figura 3. Colaboraciones entre países de publicaciones de artículos en relación a la lengua de señas de Adeyanju et al. (2021).

Acorde a los métodos de entrada de datos, se puede observar que existen diferentes tipos como se pueden observar en la Tabla 1. Brevemente, se puede mencionar que el método más recurrente es la video cámara, dada la facilidad de poder utilizarla.

Método	Ventaja	Desventaja
Cámara de video estándar	Fácil de usar; Datos de imagen 2D que son fácilmente legibles.	Diferente hardware requiere preproceso a la imagen; Ruido.
Kinect	Datos precisos; La imagen y la profundidad proporcionan más detalles a los datos; Proceso de segmentación más fácil.	Menos costoso; Sensible a la fuente infrarroja externa.
Peluquero	Multidireccional	Implementación compleja.
Conjunto de datos obtenido	Listo para usar; Viene en una gran cantidad de muestras; No se necesita hardware de entrada; Ahorra tiempo y recursos.	-
Aumento de datos	Proporcionar más datos; Evitar el sobreajuste	-

Tabla 1. Tabla de los Métodos de obtención de imágenes de Ardiansyah et al. (2021).

La mayoría de los estudios de reconocimiento de Lengua de Señas se han basado en cámara de un solo lente por su escalabilidad, y está dentro del alcance del proyecto actual de basarse solo en cámaras de un solo lente.

Al momento de obtener los datos, ya sea en video o en foto, se procede a preprocesar las imágenes, donde existen diferentes tipos de filtros y otros procesos como se muestran en la Tabla 2. No obstante, en la Figura 4 existen más métodos de preprocesamiento.

Método	Ventaja	Desventaja
Filtro gaussiano	Reduce el ruido; suaviza la imagen.	Reduce los detalles.
Filtro mediano	Filtra el ruido; Preserva características nítidas.	Sin propagación de errores.
Recorte de imágenes	Entrada consistente.	-
<i>Bootstrapping</i>	Evitar la pérdida de información; Implementación simple.	Consume Tiempo.

Tabla 2. Tabla de los Métodos de preprocesamiento de imágenes de Ardiansyah et al. (2021)

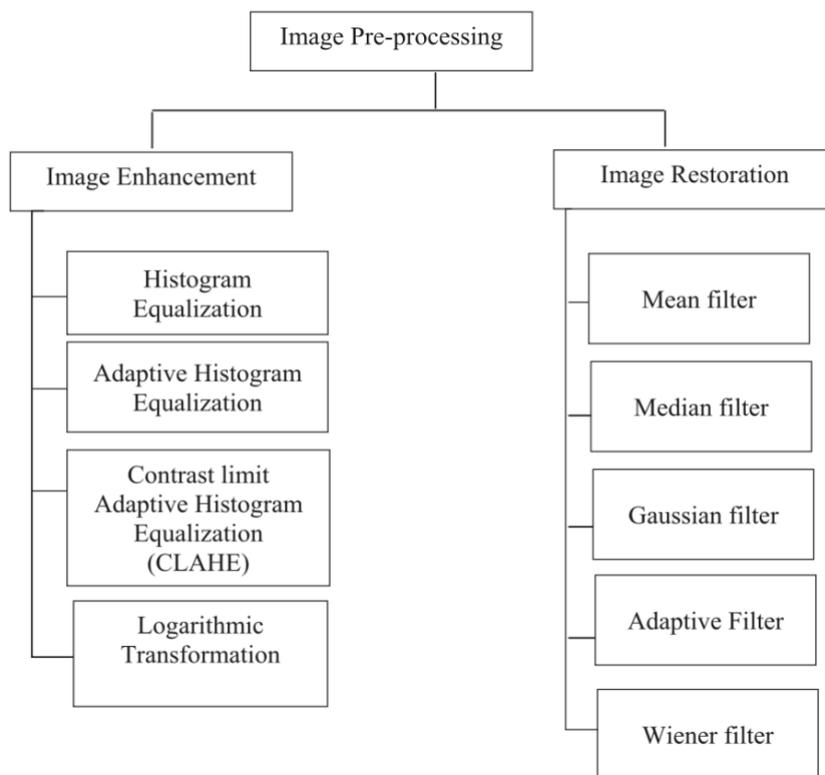


Figura 4. Diagrama de métodos de preprocesamiento de imágenes de Adeyanju et al. (2021).

Sin embargo, antes de pasar a saber cómo brindar una entrada a un modelo, se procede a que exista una segmentación adecuada de la imagen para enfocarnos en la región de interés y que por tanto tengamos las características más adecuadas a pasar al modelo. En la Tabla 3 se muestran los métodos de comparación de segmentación.

Método	Ventaja	Desventaja
Escala gris	Bajo nivel de cómputo; Implementación simple.	-
Umbral	Bajo nivel de cómputo; Rendimiento rápido.	-
Algoritmo de <i>OTSU</i>	Valor de umbral automático.	-
Filtro morfológico	Efectivo en el crecimiento de la región de intereses.	Las características pueden ser menos detalladas.
Detección de bordes Canny	Efectivo contra ruidosos y diversos ambientes.	Alto cálculo; Pérdida de tiempo.

Región sembrada creciente	Segmentación de imagen rápida; robusta y efectiva para el crecimiento de la región de intereses.	Sensible al ruido; Lento.
Borde de Sobel	Implementación simple.	Aumento del ruido.
Segmentación de la piel	Simplicidad de la implementación.	Sensible a la iluminación.
Sustracción de fondo	Bajo cálculo.	Depender de la velocidad del frame y la velocidad del objeto; Sensible a la iluminación.
Algoritmo de Viola y Jones	Eliminación de la cara.	Sensible a la condición de iluminación y la rotación invariante.

Tabla 3. Tabla de los Métodos de segmentación de Ardiansyah et al. (2021).

En la mayoría de los artículos, se aplicó el método de *Thresholding* (Ardiansyah et al., 2021), sin embargo, igual se ha observado que el escalamiento de grises es de los más utilizados antes de ingresar los datos a un modelo para disminuir el número de entradas y por tanto el uso de memoria computacional. Por otro lado, acorde a Adeyanju et al. (2021) en las últimas investigaciones, se promueve más el uso de redes neuronales y sus derivados para segmentar y obtener su caracterizaciones previo a la clasificación, entre otros métodos como se muestra en la Figura 5.

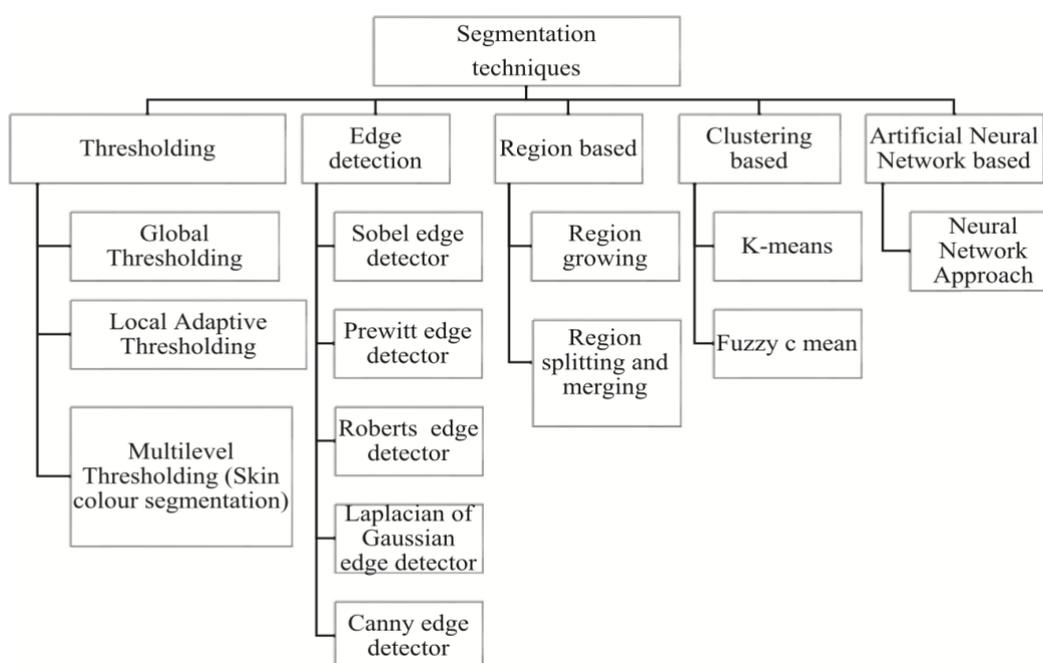


Figura 5. Diagrama de técnicas de segmentación de imágenes de Adeyanju et al. (2021).

Una vez realizada la segmentación, se realiza una extracción de características, donde algunos de los métodos de extracción de características son modelos de *machine learning* (Ver Tabla 4) como es el *Principle Component Analysis*, y sus variantes (*PCANet*), sin embargo, igual se menciona que la operación convolucional de las redes convolucionales son parte de estos métodos, pero ya pertenece a un algoritmo de clasificación.

Método	Ventaja	Desventaja
Transformada de coseno discreto	Algoritmo rápido.	Requiere pasos adicionales.
Análisis de componentes principales	Mejorar el rendimiento; Reducir el sobreajuste.	Requiere estandarización de datos.
Características robustas aceleradas	Robusto contra datos invariantes; Eficiente y más rápido en comparación con SIFT.	A veces puede ser una coincidencia falsa.
<i>PCANet</i>	Nivel de cómputo eficiente	Complejidad espacial.
Transformada de wavelet discreta	Fácil de filtrar el ruido para la señal.	Implementación compleja.

Tabla 4. Tabla de los Métodos de los métodos de extracción de características de Ardiansyah et al. (2021)

Al tener los datos de entrada ya con su caracterización, se puede proceder a modelar en un modelo de ML (*Machine Learning*) o de DL (*Deep Learning*). En la siguiente Tabla 5 se observan diferentes algoritmos de ML como de DL.

Método	Ventaja	Desventaja
Coefficiente de correlación cruzada	Bajo cálculo.	Demasiado simple para la clasificación. No aprende.
Máquinas de vectores soporte (SVM)	Memoria eficiente. Efectivo para la clasificación.	Bajo rendimiento cuando el ruido está en los datos.
Red neuronal artificial	Capaz de aprender. Red robusta tolerante a fallas.	Velocidad de convergencia lenta.
Modelo oculto de Markov (HMM)	Algoritmo de aprendizaje eficiente. Altamente preciso	Alto nivel computacional. Necesita muchos datos de capacitación.

	para la clasificación de imágenes.	
Red neuronal Convolutacional (CNN)	Altamente preciso para la clasificación de imágenes. Puede funcionar bien incluso sin segmentación o preprocesamiento.	Alto nivel computacional. Necesita hardware.
Red neuronal convolutacional con aprendizaje de transferencia	Altamente preciso para la clasificación de imágenes. Ahorra tiempo desde que está pre-entrenado.	Alto nivel computacional. Necesita hardware. El preprocesamiento necesario para adaptarse a la red.

Tabla 5. Tabla de los Métodos de clasificación de Ardiansyah et al. (2021)

Algoritmos de Aprendizaje de Máquina

En sí, se han desarrollado muchos modelos para reconocer la lengua de señas para clasificar cada seña acorde a su idioma. A principios de la década pasada, el reconocimiento de una imagen o un cuadro de un video de una señal estática ha sido respaldado por técnicas de aprendizaje automático (Adeyanju et al., 2021), como k-Nearest Neighbors, Artificial Neural Networks (ANN), máquina de vectores de soporte o *Support Vector Machine* (SVM), modelo oculto de Markov o *Hidden Markov Model* (HMM), red neuronal de convolución o *Convolutional Neural Networks* (CNN), lógica difusa y aprendizaje de conjuntos. Sin embargo, implementar dichos modelos en escenarios de la vida real no presenta la exactitud deseada. Los únicos dos modelos que han mostrado los resultados deseados para señas estáticas son SVM y CNN. Sin embargo, SVM solo como modelo puede reconocer una seña si un algoritmo o filtros extraen características antes. Por otro lado, CNN puede generar la exactitud deseada, pero a cambio requiere un alto costo computacional. (Ver Tabla 6).

Métodos de clasificación de lengua de señas	Ventajas	Desventajas
K-Cercanos Vecinos	Es fácil de implementar. Es un algoritmo simple para interpretar.	Muy sensible a las características irrelevantes. No funciona bien con un gran conjunto de datos. No funciona bien con altas dimensiones.
Redes neuronales artificiales (ANN)	Es útil donde se requiere la evaluación rápida de la función	Es computacionalmente costoso y tiene dificultades para encontrar

	objetivo aprendida. Es bastante robusto en el ruido en el conjunto de datos de entrenamiento. Tiene tolerancia a fallas.	una estructura de red adecuada.
Máquina de vectores de soporte (SVM)	Se desempeña mejor cuando se trata de multidimensiones y características continuas. Es aplicable en numerosos dominios. Tolerancia a atributos irrelevantes.	Requiere una gran muestra del conjunto de datos para lograr su máxima precisión de predicción. Los hiperparámetros a menudo son desafiantes al interpretar su impacto.
Modelo oculto de Markov (HMM)	Se desempeña relativamente bien en reconocimiento. Es más fácil de implementar y analizar. Elimina el problema de sesgo de la etiqueta.	Los HMM a menudo tienen una gran cantidad de parámetros no estructurados. Requiere una gran cantidad de capacitación para obtener mejores resultados. Requiere un gran conjunto de datos para el entrenamiento.
Convolución Network Neural (CNN)	Detecta automáticamente características importantes sin ninguna supervisión humana. Manejo de la clasificación de imágenes con éxito con alta precisión.	Alto costo computacional. Requiere muchos datos de entrenamiento para lograr una buena precisión. Falta de capacidad de ser espacialmente invariante a los datos de entrada. No codifica la posición y la orientación del objeto.
Lógica difusa	Es un sistema robusto donde no se requieren entradas precisas. Es flexible y también puede permitir modificaciones. Es una técnica de base experta que proporciona soluciones a soluciones complejas. Se trata de problemas complicados de una manera simple.	Depende completamente de la inteligencia y la experiencia humana. Tiene baja precisión, y sus predictores no siempre son correctos.
Aprendizaje Ensamblado	Mejora el rendimiento de predicción promedio. Proporciona alta precisión y un modelo más estable. Reduce la varianza de los errores predictivos.	Puede ser más difícil de interpretar. A veces, el modelo puede ser sobre ajustado o poco ajustado utilizando el método de aprendizaje del conjunto.

Tabla 6. Tabla de los Métodos de clasificación de ML de Adeyanju et al. (2021).

Algoritmos de Aprendizaje Profundo

Durante la última década, los investigadores han optimizado técnicas de aprendizaje profundo como CNN y modelos de redes neuronales recurrentes o *Recurrent Neural Networks* (RNN) para el reconocimiento de lengua de señas, incluidas las señas continuas, ya que una seña continua se puede interpretar como una secuencia de imágenes por segundo (frames por segundo - fps). En la siguiente Tabla 7, encontramos los modelos esenciales de otras investigaciones, donde modelos de CNN como *Single Shot Detector* (SSD) y MobileNetV2 pudieron obtener más del 90% de exactitud en el reconocimiento de señales estáticas en tiempo real, lo que significa que el sistema responde con una traducción de la mano o manos en el momento de recibir una entrada (Mohammedali et al., 2022):

Modelos	Tipo de seña a reconocer	Autores
VGG16	Estática	Mujahid et al., 2021
YOLOv3 & Darknet53	Estática	Mujahid et al., 2021
Single Shot Detector (SSD)	Estática	Liu et al., 2019; Mujahid et al., 2021
Capsule Networks	Estática	Bilgin & Mutludogan, 2019
MobileNetV2	Estática	Lum et al., 2020; Rathi, 2018
Inception v4 + LSTM	Continua	Bantupalli & Xie, 2019
Inception v3 + LSTM	Continua	Elhagry & Elrayes, 2021
3D CNN	Continua	Li et al., 2020
VGG16 +GRU	Continua	Li et al., 2020

Tabla 7. Modelos aplicados para el Reconocimiento de LS con CNN y RNN.

En contraste, la 3D CNN mostró la mejor exactitud para las señales continuas entre los otros modelos, pero requirió un mayor costo computacional. Luego, es esencial establecer un punto de equilibrio para aplicar los modelos en el mundo real logrando niveles de exactitud comparables a una 3D CNN mientras se mantiene una latencia equivalente o inferior a la de MobileNetV2 o SSD. Además, el entrenamiento práctico del modelo requiere la adopción de un enfoque de etiquetado automatizado para identificar y categorizar consistentemente la mano en cada frame o cuadro por segundo.

MediaPipe Holistic

La detección de la pose humana se ha convertido en un área prolífica de investigación a la luz del innovador trabajo realizado por Güler et al. (2018). Como resultado de sus esfuerzos pioneros, han surgido numerosos artículos académicos sobre el tema. Entre los algoritmos centrados en la detección de manos y las alteraciones cuando aparece el movimiento que han demostrado una mayor precisión de detección se encuentra *MediaPipe*, desarrollado por investigadores de Google (Zhang et al., 2020). *MediaPipe* es un *pipeline* de *ML* para el seguimiento de manos y reconocimiento de gestos. Donde, se busca la detección de la palma de la mano en vez de la detección directa y completa de la mano. En la investigación se menciona que se consiguió una exactitud promedio del 95.7% en la detección de la palma. Usando un *regular cross entropy loss* y *sin decoder*, brinda una base del 86.22%.

La biblioteca de *MediaPipe* enfocada en las manos está detrás de un algoritmo dividido en dos partes, la primera en detectar la palma con el algoritmo SSD modificado, adaptado y renombrado *BlazePalm Detector* (Ver Figura 6). En consecuencia, una vez que se detecta la palma de la mano, se realiza una estimación para graficar los 21 puntos de referencia de la palma más adelante (Ver Figuras 7 y 8). Cabe mencionar que periódicamente el algoritmo pregunta si continúa detectando alguna palma, logrando un menor costo computacional y baja latencia (Ver Figura 9).

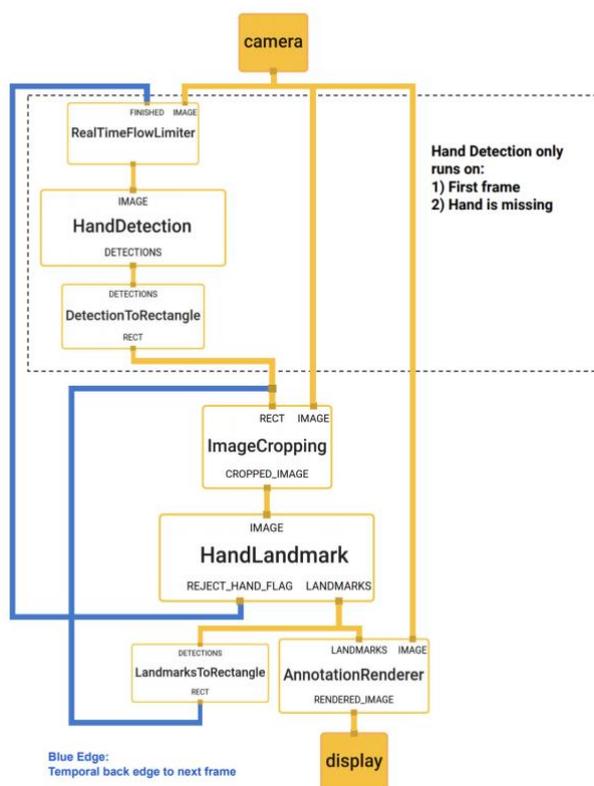
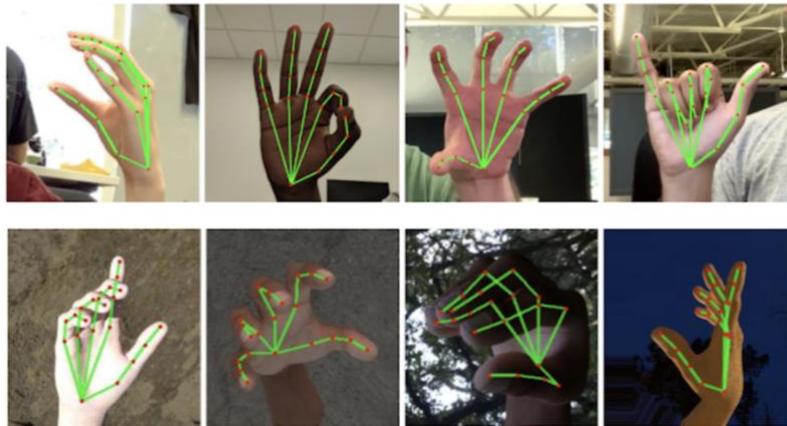
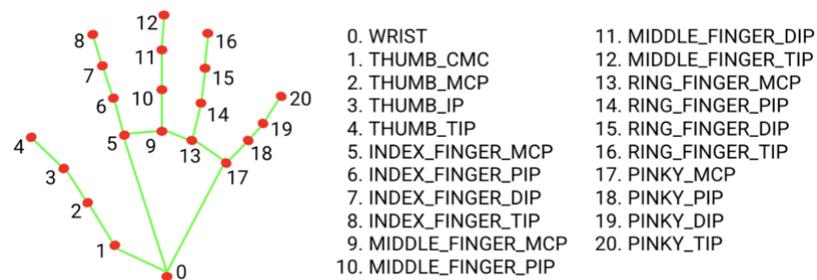


Figura 6. Diagrama del funcionamiento del algoritmo de *MediaPipe* (*BlazePalm Detector*) de Zhang et al. (2020).



Figuras 7 y 8. Marcas de las posiciones de *MediaPipe* en sus 21 puntos de las manos de Zhang et al. (2020).



Figura 9. Muestra del uso de *MediaPipe Holistic* para la detección de manos.

Reconocimiento de lengua de señas con *MediaPipe* y LSM

Como se puede observar en la Tabla 8, se compararon 14 artículos y también nuestro estudio que están relacionados con el reconocimiento de lengua de señas con *MediaPipe* y sin el. Es relevante mencionar que de estos estudios, sólo 12 (80%) crearon una nueva base de datos, 9

(60%) usaron *MediaPipe*, 6 (40%) usaron una cámara web normal y 8 (53,34%) implican LSM. Dichos artículos científicos relevantes y su comparación demuestran la brecha actual entre los algoritmos aplicados de última generación y su aplicación en LSM y otros LS con métodos y modelos de reconocimiento similares.

Otros enfoques para la lengua de señas

La traducción de la lengua de señas se considera uno de los problemas más complejos en LS debido a la baja latencia requerida para reconocer LS en un escenario real en tiempo real. Se aplicó un ejemplo de otro enfoque de un detector de poses densas con la lengua de señas alemán (Koishybay et al., 2021), que usó una señal espacio-temporal como detector de poses densas y reconocedor de palabras, y usó transformadores para traducir *Gloss2Text* (De la lengua de señas al lenguaje escrito).

Un trabajo relevante de Geng et al. (2023) pudo detectar una pose humana con señales Wifi con baja latencia porque las señales se reciben en 1D, pero desafortunadamente, la exactitud de la pose densa desarrollada fue de alrededor del 50%. No obstante, la investigación demuestra el comienzo de nuevos sensores para la pose humana en lugar de cámaras web para una mayor escalabilidad.

Referencia	Lengua(s) de seña(s)	Dispositivo	Nueva Base de Datos	Frame o Imagen del escenario	# de participantes	# de muestras	Uso de MediaPipe	# of Landmarks Used (If <i>MediaPipe</i>)	# de manos detectadas al mismo tiempo	# de clases de la dactilología	# de clases de los números (del 1 al 10)	# de clases de las palabras	Modelos entrenados para las señas estáticas	Modelos entrenados para las señas continuas	Mejor exactitud
Solís et al., 2015	Mexican	Digital Canon EOS rebel T3 EF-S 18 - 55 Camera	1	-	N.M.	N.M.	-	-	1	24	-	-	Jacobi-Fourier Moment + Multilayer Perceptron	-	95% Jacobi-Fourier Moment + Multilayer Perceptron
Solís et al., 2016	Mexicana	Cámara Digital con 4 LEDs	1	-	N.M.	N.M.	-	-	1	21	-	-	Normalized moment + Multilayer	-	93% Normalized moment + MultiLayer

		reflectores al rededor											Perceptron		Perceptron
Jimenez et al., 2017	Mexic ana	Kinect	1	-	100	1000	-	-	1	5	5	-	HAAR 2D / HAAR 3D + AdaBoost	-	95% HAAR 3D + AdaBoost
Shin et al., 2021	Ameri cana	Cámara	-	-	N.M.	Dactilol ogía LSA 780,000 Dictado con dedos A 65,774 Massey	1	21	1	26	-	-	Angle and/or Distance Features + SVM, GBL	Angle and/or Distance Features + SVM, GBL	With Angle and Distance Features + SVM: Massey dataset: 99.39% Finger Spelling A: 98.45% ASL alphabet: 87.60%

						1,815									
Indriani et al., 2021	Americana	Kinect	1	-	N.M.	900	1	21	1	-	10 (From 0 to 9)	-	Hand condition detector	Hand condition detector	95% Hand condition detector
Halder & Tayade, 2021	Americana, India, Italiana, Turques	Camera	-	-	N.M.	N.M.	1	21	1	Americana 26 India 26 Italiana 22	Americana 10 Indiana Turques 10	-	SVM, KNN, Random Forest, Decision Tree, Naive Bayes, ANN, MLP	-	99% SVM in every SL
Espejel-Cabrera et	Mexicana	Cámara con	1	1	11	2480	-	-	2	-	-	249	Decision Tree, SVM,	-	+96% SVM

al., 2021		personas con ropa oscura y fondo negro											Neural networks, Naive Bayes		
Sundar & Bagyamma 1, 2022	Americana	Cámara	1	1	4	93,600	1	21	1	26	-	-	LSTM	LSTM	99% LSTM
Chen et al., 2022	Americana	Cámara	-	-	Dataset 1: 2062 Dataset 2: 5,000 Dataset 3:	N.M.	1	21	1	21	-	-	Recursive Feature Elimination (RFE) + proposed method, CNN	-	96.3% RFE + proposed method

					12,000 Dataset 4: 300										
Samaan et al., 2022	Americana	Cámara de OPPO Reno3 Mobile Pro	1	1	5	750	1	258	2	-	-	10	-	GRU, LSTM, and BiLSTM	99% GRU
Subramanian et al., 2022	India	Cámara	1	1	N.M.	900	1	540+	2	-	-	13	-	Simple RNN , LSTM , Standard GRU, BiGRU , BiLSTM, MOPGR	99.92% MOPGRU (MediaPipe +GRU)

														U (MediaPipe +GRU)	
Mejía- Pérez et al., 2022	Mexicana	Cámara con sensor de profundidad (OAK-D)	1	1	4	3000	1	67	2	8	-	22	Gaussian Noise + RNN, LSTM, and GRU	Gaussian Noise + RNN, LSTM, and GRU	97.11% Gaussian Noise + GRU
Sosa- Jimenez et al., 2022	Mexicana	Kinect	1	1	22	18,040	-	-	2	29	10 (Del 0 al 9)	43	Hidden Markov Model (HMM)	Hidden Markov Model (HMM)	99% HMM
Rios- Figueroa et	Mexicana	Kinect	1	-	15	N.M.	-	-	-	21	-	-	Use of Spherical	-	100% Spherical and Cartesian

al., 2022													and/or Cartesian features classified by a Maximum A Posteriori (MAP)	features classified by a Maximum A Posteriori (MAP)	
Actual Estudio	Mexic ana	Cámara	1	1	11	3,677	1	42	2	29	10	-	SVM, GBL	LSTM, GRU	92% SVM 85% GRU

Tabla 8. Tabla Comparativa de la Revisión de Literatura aplicada al Reconocimiento de Lengua de Señas relacionada con MediaPipe y

sin el. Nota: N.M. = No mencionado.

Algoritmos del Estado del Arte con MediaPipe

Acorde a la Tabla 8, se pudo observar que cuando se utiliza una cámara web y se extraen las características principales, los algoritmos que más constantemente se muestran son SVM, GB, LSTM y GRU. Dichos algoritmos pertenecen a técnicas de Aprendizaje de Máquina y Aprendizaje Profundo. Por lo que, cada uno requerirá un ajuste de sus hiperparámetros o una construcción en su arquitectura acorde a las investigaciones para obtener la mayor exactitud posible al momento de reconocer las señas del LSM.

Máquina de Soporte de Vectores o Support Vector Machine (SVM)

El SVM o la Máquina de Vector de Soporte (*Support Vector Machine*) es un algoritmo de aprendizaje supervisado utilizado en problemas de clasificación y regresión. (MathWorks, 2023) El objetivo de SVM es encontrar un hiperplano que mejor separe dos clases diferentes de puntos de datos. Busca el hiperplano con el margen más amplio entre las clases, que es la anchura máxima de la región paralela al hiperplano sin puntos de datos interiores (Ver Figura 10). El SVM estándar se formula para problemas de clasificación binaria, pero los problemas multiclase se reducen a una serie de problemas binarios combinando clasificadores binarios.

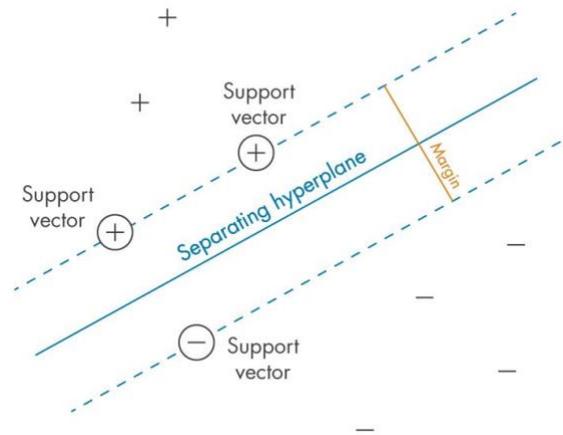


Figura 10. Muestra de definición del “margen” entre clases: el criterio que los SVM intentan optimizar de MathWorks (2023).

SVM pertenece a la clase de algoritmos de *Machine Learning* llamados métodos *kernel*, donde se utilizan funciones de *kernel* para transformar características. (Rodrigo, J., 2020) Esto permite separar clases en un espacio dimensional superior sin transformar explícitamente los datos, gracias a los métodos *kernel*. Entre los distintos tipos de *kernel*, existen: la lineal, la polinomial, la sigmoide y la RBF (Función base radial).

Asumiendo un conjunto de datos donde X es denominado:

$$(X_1 + X_1), (X_2 + X_2), \dots, (X_n + X_n)$$

Por ejemplo, $X = \{x_i, y_i\}_{i=1}^n$ donde $x_i \in \mathbb{R}^d$ y $y_i \in \{+1, -1\}$. Entrenar un SVM te permite resolver un problema de programación cuadrática como el siguiente:

$$1. \max_{\alpha_i} -\frac{1}{2} \sum_{i,j=1}^l \alpha_i y_i \alpha_j y_j K(x_i \cdot x_j) + \sum_{i=1}^l \alpha_i$$

Sujeto a: $\sum_{i=1}^l \alpha_i y_i = 0, C \geq \alpha_i \geq 0, i = 1, 2, \dots, l$

Donde $C > 0, \alpha_i = [\alpha_1, \alpha_2, \dots, \alpha_l]^T, \alpha_i \geq 0, i = 1, 2, \dots, l$, son coeficientes que corresponden a x_i, x_j con α_i no cero los cuales son llamados vectores de soporte o *Support Vectors (SV)*.

La ventaja del SVM es que es preciso en espacios de grandes dimensiones y tiene una memoria eficiente. No obstante, sus desventajas son que es propenso al ajuste excesivo, puede no contar con una estimación de probabilidad y a veces no puede ajustarse con conjuntos de datos pequeños.

Las aplicaciones del SVM incluyen reconocimiento de imagen y de voz, asignación de categoría de texto, detección de spam, análisis de los sentimientos, clasificación de la expresión génica, regresión, detección de valores atípicos y agrupamiento aplicaciones médicas de procesamiento de señales, y procesamiento del lenguaje natural.

Gradient Boosting (GB)

Acorde a Géron, A. (2019), *Gradient Boosting* es una potente técnica de aprendizaje de conjunto que ha ganado popularidad debido a su sólido rendimiento predictivo y su robustez contra el sobreajuste. Se utiliza ampliamente en diversas tareas de aprendizaje automático, incluyendo problemas de regresión y clasificación. Al combinar de forma iterativa modelos débiles, *Gradient Boosting* puede producir un modelo predictivo fuerte capaz de manejar conjuntos de datos complejos.

Cabe mencionar que, *Gradient Boosting* es otro popular algoritmo de *boosting* que funciona agregando predictores de manera secuencial a un conjunto, y cada predictor corrige los errores de su predecesor. A diferencia de *AdaBoost*, que ajusta los pesos de las instancias, *Gradient Boosting* ajusta nuevos predictores a los errores residuales cometidos por el predictor anterior.

Los modelos en la técnica de aumento de conjunto (ensemble boosting) se construyen secuencialmente mediante la minimización iterativa del error de los modelos aprendidos previamente. Esta técnica aprende un modelo predictivo combinando M modelos de árboles aditivos (f_0, f_1, \dots, f_M) para predecir los resultados (Alzamzami et al, 2020)..

$$2. f(x) = \sum_{m=0}^M f_m(x)$$

El modelo de conjunto de árboles se optimiza al reducir el error esperado de generalización L .

$$3. L = \sum_i^n (y_i - y'_i)^2$$

L es una función de pérdida que mide la pérdida delta entre el objetivo y_i y la predicción y'_i de un punto de datos.

Una implementación específica de *Gradient Boosting* para tareas de regresión utilizando Árboles de Decisión como predictores bases se llama *Gradient Tree Boosting* o *Árboles de Regresión Incrementales* (GBRT). En un ejemplo simple de regresión con GBRT, se ajusta un *DecisionTreeRegressor* al conjunto de entrenamiento, que podría ser un conjunto de entrenamiento cuadrático con ruido.

Scikit-Learn proporciona una forma conveniente de entrenar conjuntos GBRT utilizando la clase *GradientBoostingRegressor* (*Scikit-Learn*, 2023). Los estimadores basados en histogramas pueden ser órdenes de magnitud más rápidos que *GradientBoostingClassifier* y *GradientBoostingRegressor* cuando el número de muestras es mayor que decenas de miles. Además, estos estimadores tienen soporte incorporado para valores faltantes, lo que evita la necesidad de un imputador. *Scikit-Learn HistGradientBoostingClassifier* and *HistGradientBoostingRegressor*, inspirando en *LightGBM* o *GBL* (*Gradient Boost Light / Light Gradient Boosting Machine*).

Long Short-Term Memory (LSTM)

La celda *Long Short-Term Memory* (*LSTM*) fue propuesta en 1973 por Sepp Hochreiter y Jürgen Schmidhuber, y fue mejorada gradualmente a lo largo de los años por varios investigadores, como Alex Graves, Haşim Sak, Wojciech Zaremba, y muchos más.

Las celdas LSTM gestionan dos vectores de estado y, por motivos de rendimiento, se mantienen separados de forma predeterminada (Ver Figura 11).

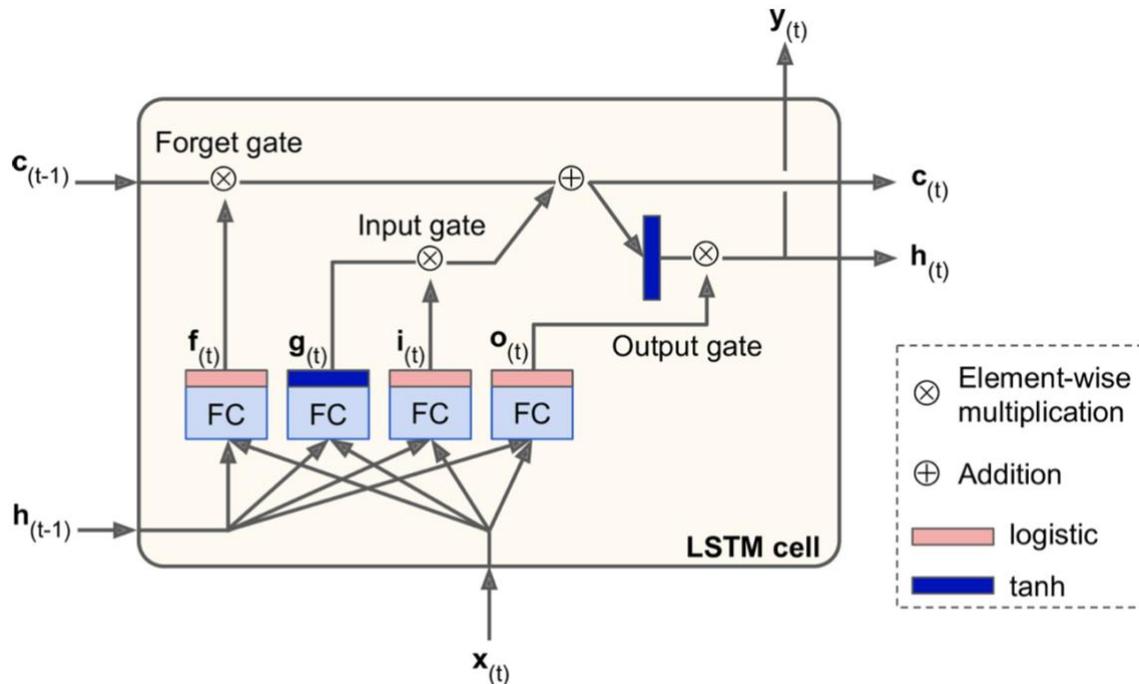


Figura 11. Celda LSTM de Géron, A. (2019).

Según Géron, A. (2019), la idea clave es que la red pueda aprender qué almacenar en el estado a largo plazo, qué desechar y qué leer de él. A medida que el estado a largo plazo c_{t-1} atraviesa la red de izquierda a derecha, puede ver que primero pasa por una puerta de olvido, eliminando algunos recuerdos y luego agrega algunos recuerdos nuevos a través de la operación de suma (que agrega las memorias que fueron seleccionadas por una puerta de entrada). El resultado c_t se envía directamente, sin más transformación. Entonces, en cada paso de tiempo, se eliminan algunos gate recuerdos y se agregan otros. Además, después de la operación de suma, el estado a largo plazo se copia y pasa a través de la función \tanh , y luego el resultado es filtrado por la puerta de salida. Esto produce el estado a corto plazo h_t (que es igual a la salida de la celda para este paso de tiempo y_t).

Primero, el vector de entrada actual x_t y el estado anterior a corto plazo h_{t-1} se alimentan a cuatro capas diferentes completamente conectadas. Todos tienen un propósito diferente:

- La capa principal es la que genera g_t . Tiene la función habitual de analizar las entradas actuales x_t y el estado anterior (a corto plazo) h_{t-1} . En una celda básica, no hay nada más que esta capa, y su salida va directamente a y_t y h_t . Por el contrario, en una celda LSTM, la salida de esta capa no sale directamente, sino que se almacena parcialmente en el estado a largo plazo.
- Las otras tres capas son controladores de puerta. Dado que utilizan la función de activación logística, sus salidas van de 0 a 1. Como puede ver, sus salidas se alimentan a operaciones de multiplicación por elementos, por lo que, si emiten 0, cierran la puerta, y si emiten 1, abren la puerta. Específicamente:

- La puerta de olvido (controlada por f_t) controla qué partes del estado a largo plazo deben borrarse.
- La puerta de entrada (controlada por i_t) controla qué partes de $g(t)$ deben agregarse al estado a largo plazo (por eso está "almacenado parcialmente").
- Finalmente, la puerta de salida (controlada por o_t) controla qué partes del estado a largo plazo deben leerse y emitirse en este paso de tiempo (tanto para h_t como para y_t).

En resumen, una celda LSTM puede aprender a reconocer una entrada importante (ese es el papel de la puerta de entrada), almacenarla en el estado a largo plazo, aprender a preservarla durante el tiempo que sea necesario (ese es el papel de la puerta de olvido), y aprender a extraerlo cuando sea necesario. Esto explica por qué han tenido un éxito sorprendente al capturar patrones a largo plazo en series de tiempo, textos largos, grabaciones de audio y más.

Las siguientes ecuaciones resumen cómo calcular el estado a largo plazo de la celda, su estado a corto plazo y su salida en cada paso de tiempo para una sola instancia:

$$\begin{aligned}
 4. \quad i_t &= \sigma (W_{xi}^T \cdot x_t + W_{hi}^T \cdot h_{t-1} + b_i) \\
 5. \quad f_t &= \sigma (W_{xf}^T \cdot x_t + W_{hf}^T \cdot h_{t-1} + b_f) \\
 6. \quad o_t &= \sigma (W_{xo}^T \cdot x_t + W_{ho}^T \cdot h_{t-1} + b_o) \\
 7. \quad g_t &= \tanh (W_{xg}^T \cdot x_t + W_{hg}^T \cdot h_{t-1} + b_g) \\
 8. \quad c_t &= f_t \otimes c_{t-1} + i_t \otimes g_t \\
 9. \quad y_t = h_t &= o_t \otimes \tanh (c_t)
 \end{aligned}$$

- W_{xi} , W_{xf} , W_{xo} , W_{xg} , son las matrices de peso de cada una de las cuatro capas para su conexión con el vector de entrada $\mathbf{h}(t)$.
- W_{hi} , W_{hf} , W_{ho} , and W_{hg} , son las matrices de peso de cada una de las cuatro capas para su conexión con el estado de corto plazo anterior $\mathbf{h}(t-1)$.
- b_i , b_f , b_o , and b_g , son los términos de sesgo para cada una de las cuatro capas.

Gated Recurrent Unit (GRU)

Kyunghyun Cho et al. propusieron la celda *de Gated Recurrent Unit (GRU)* en un artículo de 2014 que también presentó la red Codificador-Decodificador. La celda GRU es una versión simplificada de la celda LSTM y parece funcionar igual de bien (Ver Figura 12). Las principales simplificaciones son (Géron, A., 2019):

- Ambos vectores de estado se fusionan en un solo vector \mathbf{h}_t .
- Un único controlador de puerta controla tanto la puerta de olvido como la puerta de entrada. Si el controlador de puerta emite un 1, la puerta de olvido está abierta y la puerta de entrada está cerrada. Si da como resultado un 0, sucede lo contrario. En otras palabras, cada vez que se debe almacenar una memoria, primero se borra la

ubicación donde se almacenará. Esta es en realidad una variante frecuente de la celda LSTM en sí misma.

- No hay puerta de salida; el vector de estado completo se emite en cada paso de tiempo. Sin embargo, hay un nuevo controlador de puerta que controla qué parte del estado anterior se mostrará a la capa principal.

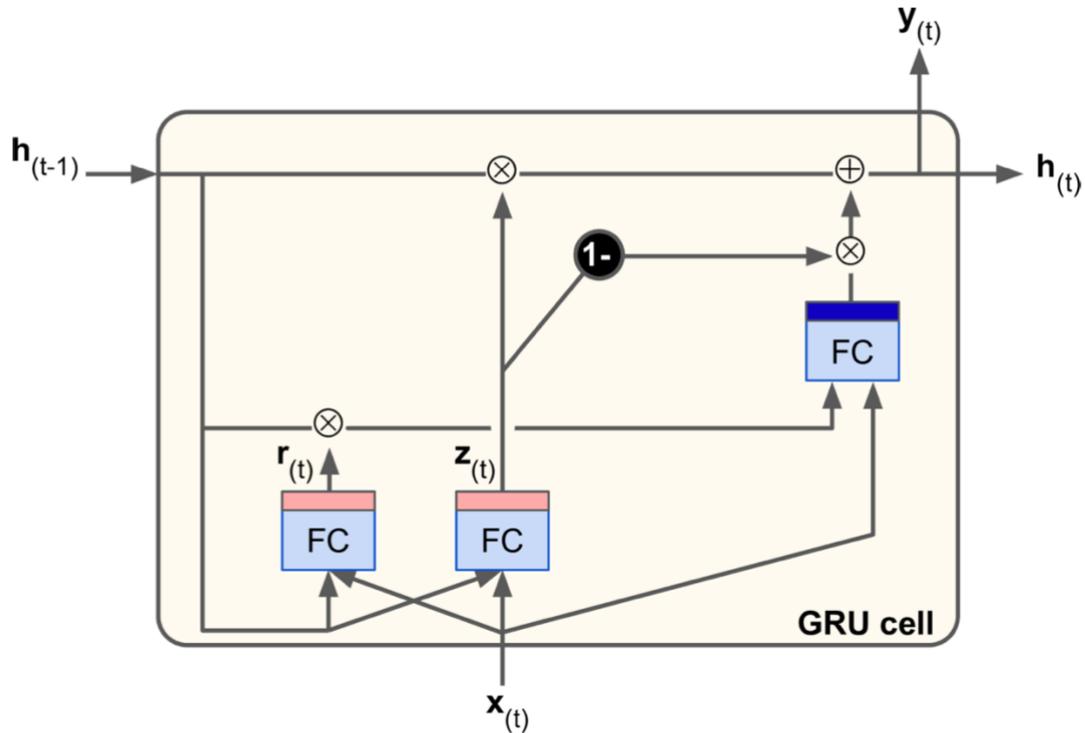


Figura 12. Celda GRU de Géron, A. (2019).

Las celdas de LSTM o GRU son una de las principales razones del éxito de las RNN en los últimos años, en particular para aplicaciones en procesamiento de lenguaje natural (NLP).

Las siguientes ecuaciones resumen cómo calcular el estado de la celda de GRU en cada paso de tiempo para una sola instancia:

$$\begin{aligned}
 10. z_t &= \sigma (W_{xz}^T \cdot x_t + W_{hz}^T \cdot h_{t-1} + b_z) \\
 11. r_t &= \sigma (W_{xr}^T \cdot x_t + W_{hr}^T \cdot (r_t \otimes h_{t-1}) + b_g) \\
 12. g_t &= \tanh (W_{xg}^T \cdot x_t + W_{hg}^T \cdot h_{t-1} + b_g) \\
 13. h_t &= z_t \otimes h_{t-1} + (1 - z_t) \otimes g_t
 \end{aligned}$$

Metodología

Metodología de investigación

Para realizar la presente investigación, se plantea una adaptación de la metodología de investigación de Sampieri et al. (1997), y los pasos de reconocimiento de señas de la revisión literaria de Ardiansyah et al. (2021). Por lo que se genera el siguiente diagrama (Ver Figura 13) que tiene los siguientes pasos de manera secuencial.

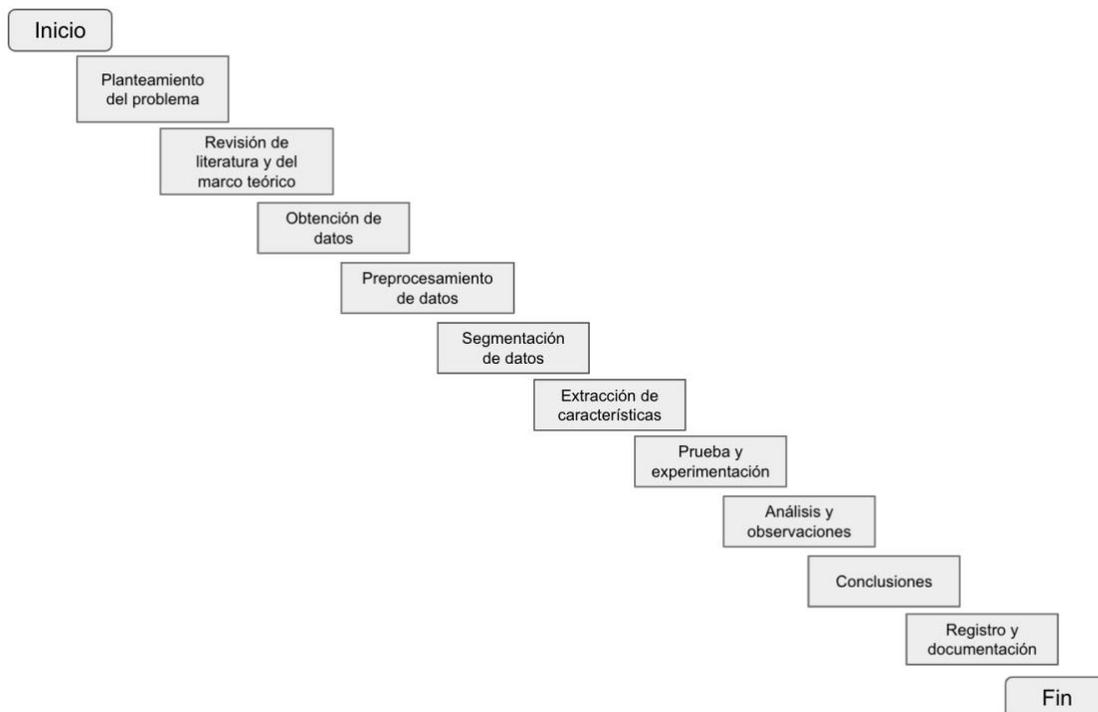


Figura 13. Propuesta de metodología de investigación. Elaboración propia, basada en Sampieri et al. (1997) y Ardiansyah et al. (2021)

Una breve explicación del mismo diagrama, sería la siguiente: Primeramente, se empieza con el planteamiento del problema para definir qué problema se pretende resolver, después se procede a revisar la literatura, y el estado del arte. Posteriormente, se pasa a obtener los datos necesarios para analizar, que en este caso sería imágenes/videos, luego su preprocesamiento, su segmentación y por último su extracción de características para que puedan entrar los datos como input a los modelos de aprendizaje donde se pondrán a prueba y se experimentarán con ellos. Luego, de obtener resultados de los modelos, se analizarán sus estadísticos y se harán conclusiones sobre los mismos para realizar la documentación necesaria acorde al plan de la tesis.

Propuesta de metodología de experimentación

En este apartado adaptamos la metodología general (Adeyanju et al., 2021) con *MediaPipe* ya que con ella podíamos extraer las características sin preprocesar con ningún filtro excepto los incluidos dentro del algoritmo y clasificarlas posteriormente. (Ver Figura 14).

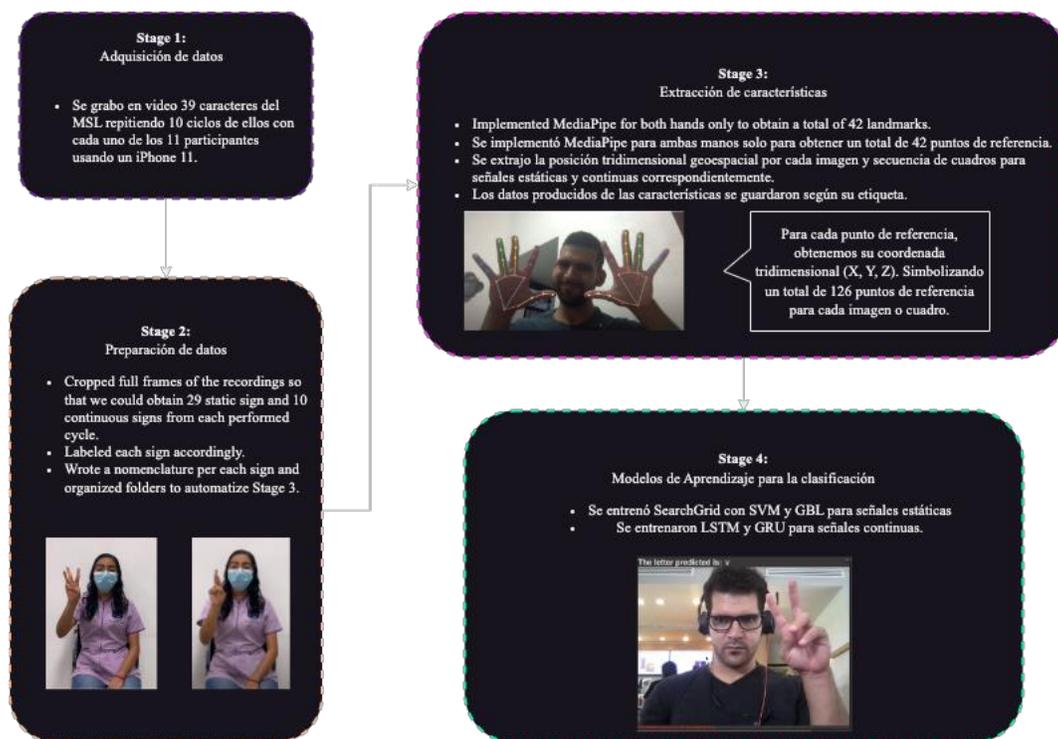


Figura 14. Proceso adaptado para el reconocimiento de lengua de señas.

Adquisición de datos

Creamos una base de datos de LSM que contiene su dactilología (29 caracteres) y los diez primeros números del uno al diez. La base de datos se creó a partir de grabaciones producidas por una vez hablantes del LSM, quienes produjeron diez series de lengua de señas y números. Las diez series se realizaron con ambas manos (cinco con la mano derecha y cinco con la mano izquierda para aumentar los datos y evitar el sobreajuste). (Rodríguez et al., 2023a, 2023b, 2023c, 2023d).

El aprendizaje de los hablantes de LSM provino de diferentes fuentes:

- Dos hablantes nativos son personas sordas que aprendieron LSM para comunicarse.
- Dos hablantes bilingües que aprendieron LSM de sus padres.
- Siete oradores que aprendieron LSM en un programa educativo.

El dispositivo de captura del conjunto de datos fue un iPhone 11 modelo MWM32LZ/A, y la configuración de grabación fue en 4k con una resolución de 3.840 x 2.160. Todas las

muestras fueron grabadas con 30 FPS (Frames Por Segundo) cuando el iPhone estaba horizontal y balanceado con un trípode (Ver Figura 15).



Figura 15. Ejemplo de toma de la señal Número 2 (Señal Estática).

Preparación de datos

Junto a un equipo de dos voluntarios de servicio social de la Universidad Autónoma del Estado de Morelos (UAEM), se capturaron con precisión las grabaciones de los ciclos de cada grabación de señales estáticas y continuas. Posteriormente, los videos se editaron para que las señales continuas tuvieran los tiempos de inicio y finalización apropiados para cada señal continua y una captura de pantalla para cada señal estática. Cabe mencionar que, en su mayor parte, cada vídeo dura un segundo en promedio.

Además, se etiquetaron señales estáticas en formato PASCAL VOC en XML con LabelImg para futuros investigadores que quisieran realizar otros experimentos. (Rodríguez et al., 2023g, 2023e).

Extracción de características

Para extraer las características de cada imagen de las muestras de señales estáticas y fotogramas de video completos de las muestras de señales continuas, se utilizó *MediaPipe* para extraer los 21 puntos de referencia de cada mano; es decir, se extrajeron 42 puntos de referencia en total por imagen o fotograma. Vale la pena mencionar que cada hito tiene tres puntos de datos relevantes, que son las coordenadas tridimensionales (X, Y, Z) que se recopilaron. Se extrajeron ciento veintiséis puntos de datos de coordenadas de los 42 puntos de referencia al final de cada imagen o fotograma.

Modelos de aprendizaje para la clasificación

Una vez que se extrajeron todas las coordenadas tridimensionales, se aplicaron SVM y *Gradient Boost Light* o *Light Boosting Gradient* (GBL) para entrenar con una cuadrícula de búsqueda de signos estáticos. Por otro lado, para las señales continuas se aplicaron los modelos *Long Short-Term Memory* (LSTM) y *Gradient Recurrent Unit* (GRU) con tres capas cada uno. (Rodríguez et al., 2023f).

Métricas de evaluación

Utilizamos múltiples métricas, como exactitud, precisión, recuperación y puntuación F1, para evaluar el rendimiento de nuestros modelos. Dado que nuestro estudio implica una clasificación multiclase, empleamos un macro promedio para calcular la precisión, la recuperación y la puntuación F1. Como Grandini et al. (2020), el macro promedio (M_{avg}) implica calcular la media general de cada clase k .

Este enfoque utiliza numeradores que van de cero a uno, lo que permite una evaluación integral del total de verdaderos positivos, verdaderos negativos, falsos negativos y falsos positivos de cada métrica.

La exactitud (A) se refiere a la proporción de predicciones correctas que hace un modelo. Esta medida se determina calculando el número de verdaderos positivos (T_p) y verdaderos negativos (T_n) y luego dividiendo este valor por la suma de verdaderos positivos (T_p) y verdaderos negativos (T_n), además de la suma de falsos positivos (F_p) y falsos negativos (F_n).

$$14. A = \frac{T_p + T_n}{T_p + T_n + F_p + F_n}$$

La precisión (P) se refiere a una métrica estadística que representa la proporción de verdaderos positivos (T_p) a la suma de verdaderos positivos (T_p) y falsos positivos (F_p). La precisión se determina utilizando esta misma fórmula cuando se trabaja con datos ponderados. (Macro promedio de Precisión - M_{avgP}).

$$15. P = \frac{T_p}{T_p + F_p}$$

$$16. M_{avgP} = \frac{\sum_{k=1}^K P_k}{K}$$

Recall (R) se refiere a una medida que calcula la proporción de verdaderos positivos (T_p) a la suma de verdaderos positivos (T_p) y falsos negativos (F_n). Esta fórmula sigue siendo

aplicable cuando se trabaja con datos ponderados (Macro promedio de *Recall* - M_{avgR}).

$$17. R = \frac{T_p}{T_p + F_n}$$

$$18. M_{avgR} = \frac{\sum_{k=1}^K R_k}{K}$$

$M_{avgF1-score}$ es una métrica estadística que proporciona una media ponderada de la medida *F1 - Score*. Los pesos se basan en la probabilidad de clase, y a cada clase se le asigna un peso de probabilidad.

$$19. M_{avgF1-score} = 2 \left(\frac{M_{avgP} * M_{avgR}}{M_{avgP}^{-1} + M_{avgR}^{-1}} \right)$$

Experimentación y resultados

Bases de datos

Realizamos nuestro experimento utilizando una base de datos creada por nosotros mismos con el conjunto de datos de dactilología de LSM y sus números. Todos los conjuntos de datos y la generación de sus resultados se almacenaron en un repositorio de Google Drive para usar Google Collab, y luego se publicaron en Mendeley Data.

A continuación, se presentan las siguientes 7 bases de datos liberadas con sus DOIs respectivos en formato APA. Se puede copiar y pegar el link completo del doi para dirigirse a la página de Mendeley Data donde se publicó y que contiene más detalle al respecto:

- Rodriguez, M., Outmane, O., Bassam, A., & Noureddine, L. (2023a). Mexican Sign Language's Dactylogy and Ten First Numbers -Raw videos. From person #1 to #3 [Data set]. Mendeley. V1. <https://doi.org/10.17632/48XYBSMVPV.1>
- Rodriguez, M., Outmane, O., Bassam, A., & Noureddine, L. (2023b). Mexican Sign Language's Dactylogy and Ten First Numbers -Raw videos. From person #4 to #7 [Data set]. Mendeley. V1. <https://doi.org/10.17632/JP4YMF2VJW.1>
- Rodriguez M., Outmane, O., Bassam, A., & Noureddine, L. (2023c). Mexican Sign Language's Dactylogy and Ten First Numbers -Raw videos. From person #8 to #10 [Data set]. Mendeley. V1. <https://doi.org/10.17632/69FMDB25XM.1>
- Rodriguez, M., Outmane, O., Bassam, A., & Noureddine, L. (2023d). Mexican Sign Language's Dactylogy and Ten First Numbers -Raw videos. Person #11 [Data set]. Mendeley. V1. <https://doi.org/10.17632/3DZN5RSTWX.1>

- Rodriguez, M., Outmane, O., Bassam, A., & Noureddine, L. (2023e). Mexican Sign Language's Dactylology and Ten First Numbers—Extracted Features and Models [Data set]. Mendeley. V1. <https://doi.org/10.17632/HMSC33HMKZ.1>
- Rodriguez, M., Outmane, O., Bassam, A. & Noureddine, L. (2023f). Mexican Sign Language's Dactylology and Ten First Numbers—Labeled images and videos. From person #1 to #5 [Data set]. Mendeley. V1. <https://doi.org/10.17632/5S4MT7XRD9.1>
- Rodriguez, M., Outmane, O., Bassam, A., & Noureddine, L. (2023g). Mexican Sign Language's Dactylology and Ten First Numbers—Labeled images and videos. From person #6 to #11 [Data set]. Mendeley. V1. <https://doi.org/10.17632/67HTNZMWBB.1>

Las bases de datos anteriores son videos sin editar, videos e imágenes editados y etiquetados, las coordenadas de cada seña, y por último están los pesos de los modelos entrenados.

Detalles de implementación

Se pretendía tener 110 muestras por personaje, pero se recuperaron 90-100 de los videos grabados debido a la transferencia de datos. (Tablas 9 y 10).

Seña estática		
Caracter/ Número	Label	# de muestras
1	0	92
2	1	92
3	2	92
4	3	92

5	4	92
6	5	92
7	6	92
8	7	92
a	8	94
b	9	94
c	10	94
d	11	94
e	12	94
f	13	94
g	14	94
h	15	94
i	16	93
l	17	93
m	18	93
n	19	93
o	20	93
p	21	93
r	22	93

s	23	93
t	24	93
u	25	92
v	26	93
w	27	93
y	28	93
Total	-	2696

Tabla 9. Conjunto de datos etiquetados y el número de muestras para las señas estáticas.

Seña continua		
Caracter/ Número	Label	# de muestras
9	0	100
10	1	100
j	2	102
k	3	102
ll	4	86
ñ	5	101
q	6	88

rr	7	101
x	8	101
z	9	100
Total		981

Tabla 10. Conjunto de datos etiquetados y el número de muestras para las señas continuas.

Durante la extracción de puntos de referencia, los videos de menos de 1 segundo copiaron datos de su último cuadro hasta que se recopilaron 30 cuadros de datos. Mientras que para los que superaron el segundo, solo se cortó a los primeros 30 fotogramas. El código de Renotte (2020) se utilizó como línea de base y se adaptó de acuerdo con el conjunto de datos y las necesidades del modelo.

Configuración de los modelos

El porcentaje de datos utilizados para cada etapa del modelado fue el siguiente: 80% para entrenamiento, 10% para prueba y 10% para validación, donde se utilizó la versión gratuita de Google Collab.

Para las señas estáticas, SVM y GBL se aplicaron con GridSearch (Ver Tabla 11). Por otro lado, para las señas continuas, Tensorflow y Keras se usaron para aplicar LSTM y GRU (Ver Figura 16) donde se implementaron inicializadores "*he normal*" y funciones de activación "*elu*" en lugar de inicializadores aleatorios y funciones de activación "*relu*" correspondientes. porque la función de activación "*elu*" ha obtenido una convergencia más rápida en otros estudios (Géron, 2019). No obstante, la única excepción para ambos modelos fue la última capa densa que usó una función de activación "*softmax*" para la clasificación con un inicializador de kernel "*glorot uniform*" y una "*L2*" como regularizador de *kernel*. Además, el modelo de GRU utilizó un *Dropout* de 0,2 en ambas capas. Además, el número de épocas para LSTM y GRU fue de 350 y 1200, respectivamente. Por último, los modelos LSTM y GRU obtuvieron correspondientemente 203,690 parámetros (todos entrenables) y 146,602 parámetros (sólo 62 no entrenables) (Para más referencia ver Apéndice 1).

De la Figura 16, se puede ver una representación en imagen de las capas. Es relevante mencionar que el número que menciona cada número por capa es el tamaño de la entrada recibida. Sin embargo, solo *MediaPipe* puede recibir otras entradas diferentes, que luego se pueden adaptar dentro de sus capas debido a la calidad de la cámara y otros factores.

Model	Grid Search Params	Params	Best Params
SVM	C	0.1, 1, 10, 100, 800, 1000, 1200	1200
	Gamma	1, 0.1, 0.01, 0.001, 0.0001	0.1
	Kernel	rbf, linear, poly, sigmoid	poly
	Cross Validation	5	5
GBL	Loss	Categorical Crossentropy	Categorical Crossentropy
	Learning Rate	0.09, 0.1, 0.11, 0.125, 0.15	0.11
	Max Iter	100, 150, 200, 250, 300, 350	250
	Cross Validation	3	3

Tabla 11. Parámetros para señas del LSM estáticas

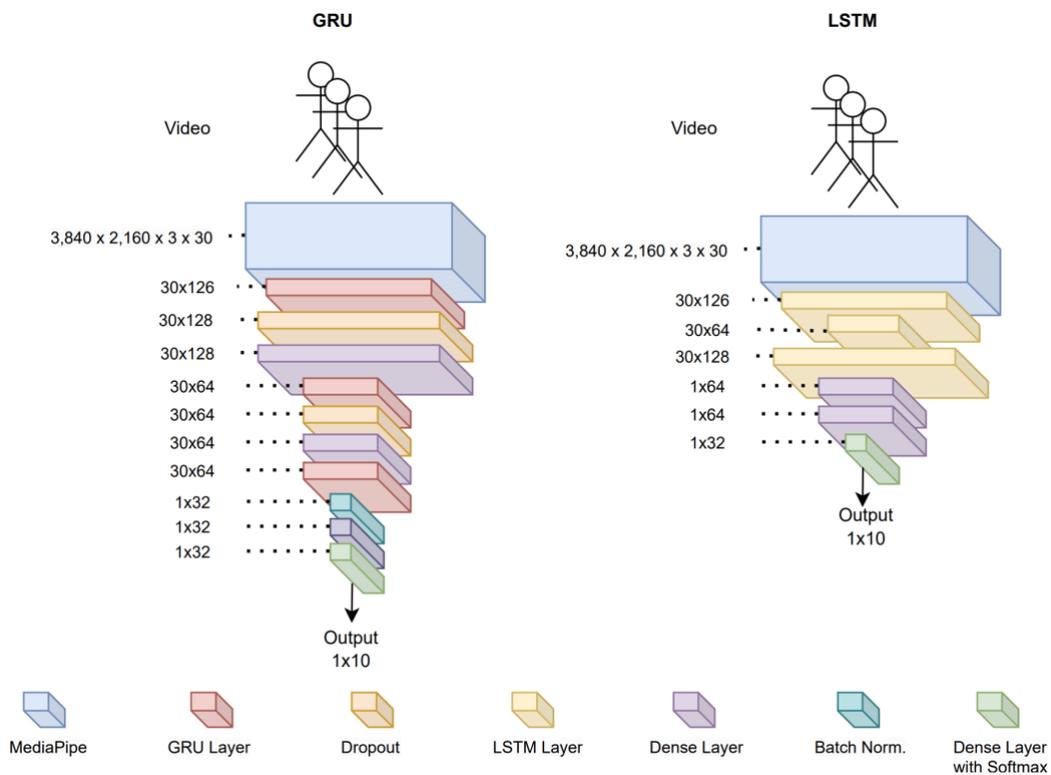


Figura 16. Arquitectura de GRU y LSTM

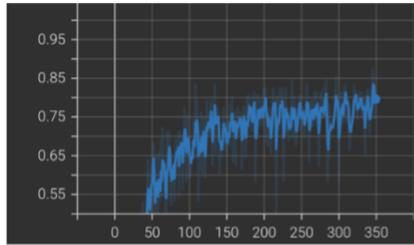
Análisis cuantitativo

Como se puede ver en la siguiente Tabla 12, SVM supera a GBL en todas las métricas en ambos conjuntos de datos (Prueba y Validación) con una precisión de +91 %. Mientras tanto, GRU supera a LSTM en todas las métricas en ambos conjuntos de datos (Prueba y Validación) con una precisión de +81 %.

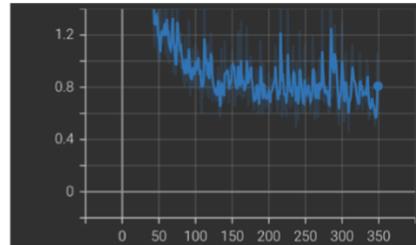
Modelos	Fase	Exactitud	Precisión (Macro)	Recall (Macro)	F1 Score (Macro)
GBL	Testeo / Testing	77,04%	76,85%	76,82%	76,19%
SVM*		94,07%	93,73%	94,25%	93,56%
LSTM		79,59%	80,31%	80,93%	80,28%
GRU**		84,69%	86,01%	86,00%	85,57%
GBL	Validación	77,04%	77,28%	76,83%	76,15%
SVM*		91,48%	90,93%	89,96%	89,90%
LSTM		76,77%	75,87%	77,46%	76,00%
GRU**		81,82%	82,50%	82,19%	81,47%

Tabla 12. Métricas de evaluación. Nota: *Modelo con mejor desempeño para Señas Estáticas. **Modelo con mejor desempeño para Señas Continuas.

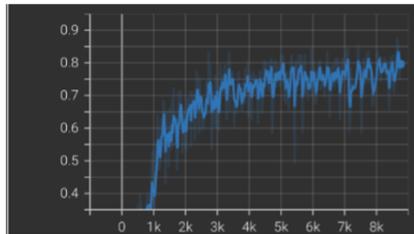
En las siguientes figuras 17 y 18, los gráficos de exactitud y pérdida se compararon entre los modelos LSTM y GRU durante las pruebas y la validación. Se puede ver que los gráficos GRU son más suaves en comparación con los gráficos LSTM. De ello se puede decir que el modelo GRU en esta situación y problema es más propenso a aprender a reconocer la lengua de signos continua que el modelo LSTM.



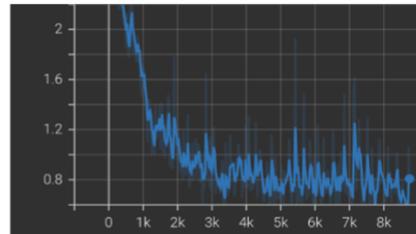
a) Exactitud categórica de época durante la prueba de LSTM



b) Pérdida de época durante el testeo de LSTM

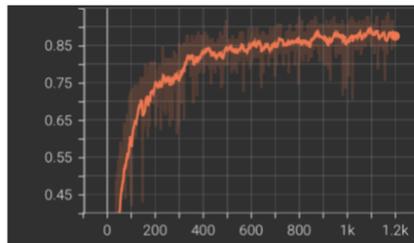


c) Exactitud categórica de evaluación durante la validación de LSTM

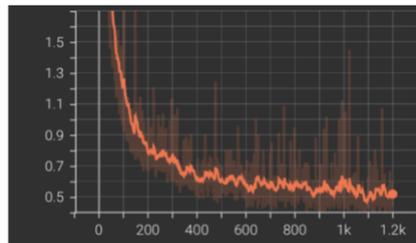


d) Pérdida de evaluación durante la validación de LSTM

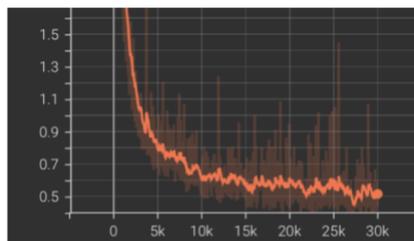
Figura 17. Gráficos de pérdida de época y exactitud categórica de LSTM.



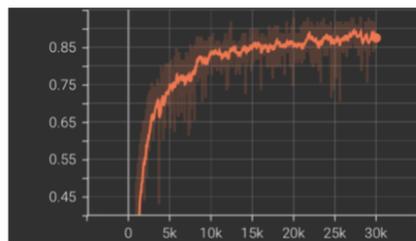
a) Exactitud categórica de época durante las pruebas de GRU



b) Pérdida de época durante el testeo de GRU



c) Exactitud categórica de evaluación durante la validación de GRU



d) Pérdida de evaluación durante la validación de GRU

Figura 18. Gráficos de pérdida de época y exactitud categórica de GRU.

Análisis comparativo

Las matrices de confusión GBL y LSTM arrojaron más negativos positivos que SVM y GRUM (Ver Apéndice 2). Además, la exactitud categórica y las curvas de pérdida en las pruebas y validación de LSTM y GRU demostraron que GRU tiene un reconocimiento más estable en el momento del entrenamiento. Por lo tanto, SVM y GRU son más exactos en el reconocimiento múltiple de signos de LSM.

Todos los modelos se almacenaron en el mismo repositorio con la biblioteca joblib. Solo GRU y LSTM se guardaron manualmente con formatos "h5". Además, se implementó una implementación dentro de Google Colab usando los modelos con Javascript (Ver Figura 19), donde se adaptó el código de TheAIGuysCode, (2020) para usar una cámara web en Google Collab y aplicar los modelos entrenados correspondientemente (Rodriguez et al., 2023h).

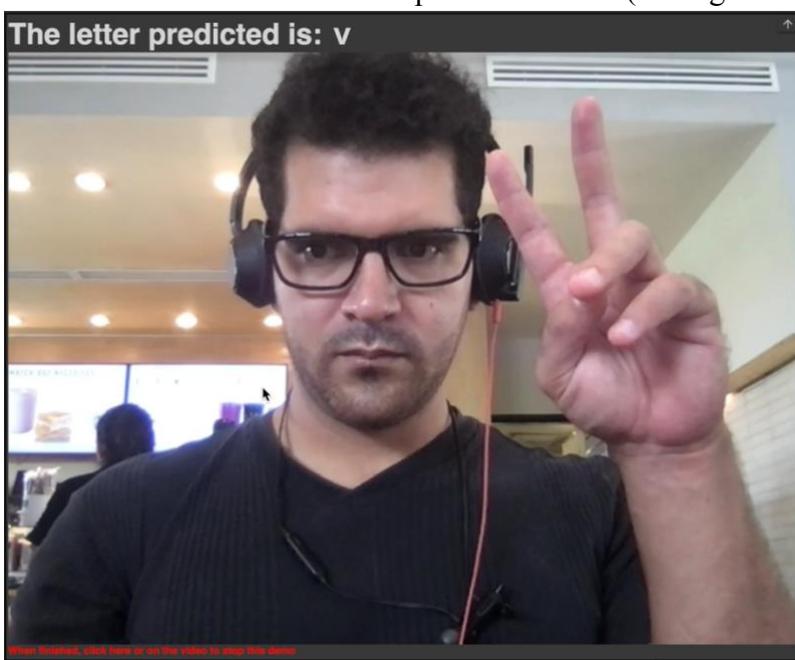


Figura. 19. Ejemplo de la letra “V” Reconocida.

Discusión

Mediapipe nos ayudó a recopilar los puntos de referencia esenciales de las manos con gran precisión y realizar la detección en tiempo real. Además, podría darnos los puntos de referencia de todo el cuerpo y de la cara si así lo deseáramos. Sin embargo, la investigación actual solo se centró en la dactilología de LSM y los primeros diez dígitos, por lo que nos centramos solo en lo necesario. Además, creemos que se desarrollarán otros algoritmos de pose densa y diferentes sensores que podrían reemplazar a *Mediapipe* y la cámara web si los resultados superan las métricas de vanguardia actuales.

Es relevante mencionar que a pesar de que para el entrenamiento se utilizó una cámara de calidad como la del iPhone 11, se pueden utilizar otras cámaras de menor calidad para aplicar el mismo modelo cuando se utiliza *MediaPipe*. Además, dado que se registró un escenario completo desde la cadera hasta la cabeza y dos manos, los modelos aplicados se entrenaron respectivamente con estos escenarios reales.

Por lo tanto, se vio que SVM y GRU superaron a GBL y LSTM correspondientemente en cada métrica de evaluación. Sin embargo, para señales estáticas, SVM ha mostrado una precisión excelente (+90 %) incluso con unas pocas muestras (<100 por señal). En línea con otros estudios, una precisión tan positiva demuestra que SVM (Halder & Tayade, 2021) tiene el potencial de reconocer signos estáticos cuando se extraen primero las características (puntos de referencia en este caso). Entonces, dado que GBL obtuvo una precisión menor que SVM, es posible que necesite más muestras para obtener un modelo más exacto y evitar el sobreajuste y el ajuste insuficiente.

Al implementar los modelos en tiempo real, se observó que dado que los modelos se entrenaban con ambas manos pero casi siempre una mano descansaba alrededor de la cintura, se detectó una imprecisión de reconocimiento en los casos en que solo se mostraba una mano y le faltaba la otra, o ambas manos se encontraban, pero una mano no estaba en reposo. Tales casos nos recuerdan que necesitaremos aún más muestras para el aumento de datos y consideraciones para otras características durante el entrenamiento, podría ser que necesitemos alrededor de cuatro a cinco veces más datos si lo comparamos con otro estudio relacionado con LSM (Sosa-Jimenez et al., 2022).

El estudio actual no se centró en el reconocimiento de palabras o la traducción de oraciones debido a la falta de conjuntos de datos de código abierto y la limitación de tiempo en LSM. Sin embargo, en caso de que se desee desarrollar más investigación en términos de LSM, se creará un conjunto de datos similar como PHOENIX14 (Camgoz et al., 2020; Min et al., 2021) de la lengua de señas alemana con diferentes temas y contextos en las conversaciones.

Conclusiones y limitaciones

Los resultados de este trabajo de investigación proporcionan una adición significativa y notable al dominio del reconocimiento de LSM. La importancia de esta contribución radica en su potencial para ayudar a las personas con problemas de comunicación en México y los investigadores que trabajan en este campo. A pesar de la investigación significativa y los prototipos y modelos tecnológicos para respaldar los sistemas de comunicación pública sin intérpretes humanos, la mayoría de los esfuerzos se han orientado hacia la lengua de señas americana (LSA) u otras lenguas de señas. El presente trabajo presenta una base de datos de código abierto de signos y numeración LSM, que comprende 29 caracteres de dactilología de la lengua de señas mexicana y sus primeros diez dígitos.

Se empleó *MediaPipe* para la extracción manual de características y los patrones de señales se reconocen con precisión mediante SVM y GRU, logrando una precisión del 92% y el 86% para señales estáticas y continuas, respectivamente. Los resultados demuestran las perspectivas prometedoras de las técnicas de aprendizaje automático con DensePose, como *MediaPipe* con puntos de referencia, en el reconocimiento de LSM y su capacidad para facilitar el desarrollo de sistemas de comunicación más inclusivos que atiendan las necesidades de las personas con discapacidades de comunicación en México.

Los resultados de este estudio sirven como inspiración para futuros investigadores, lo que impulsa una mayor investigación sobre varios aspectos, incluido el reconocimiento de otras lenguas de señas y el desarrollo de conjuntos de datos de fuente abierta. Al utilizar la metodología investigada y aplicada descrita en este estudio, los investigadores pueden encontrar orientación e inspiración para reconocer otras lenguas de señas en función de su dactilología y palabras, así como apoyar los esfuerzos continuos en la traducción a nivel de oración.

Vale la pena señalar que la consideración del estudio de equipos de bajo costo garantiza la escalabilidad, lo que permite una mayor accesibilidad y una adopción potencial por parte de los usuarios finales y las instituciones. Esta escalabilidad facilita aún más el uso y la implementación de la tecnología de reconocimiento de la LSM, haciéndola más factible para aplicaciones prácticas.

En conclusión, esta investigación contribuye significativamente al avance del reconocimiento de LSM al proporcionar una base de datos de código abierto, emplear técnicas efectivas de aprendizaje automático y aprendizaje profundo e inspirar futuros esfuerzos de investigación. Las implicaciones de este trabajo se extienden a mejorar la comunicación y la inclusión de las personas con discapacidades de la comunicación en México, al mismo tiempo que proporciona información valiosa para los investigadores en el campo.

Resumen de contribuciones

- La investigación actual demuestra la viabilidad de la implementación en tiempo real de algoritmos computacionales de bajo costo para el reconocimiento estático y continuo de la de señas en el LSM que se investigaron en el estado del arte, incluida la dactilología completa del LSM (Serafín, M. E., & González, R., 2011). con “LL” y “RR”, y los diez primeros números. El equipo de investigación abordó la falta de una base de datos pública LSM y de código abierto creando una ahora disponible para futuras investigaciones científicas abierta a la comunidad científica, lo cual es un paso crucial para fomentar la innovación en este campo.
- Uno de los aspectos más destacables de este estudio es el uso de escenarios de video completos y precisos para desarrollar el reconocimiento de la lengua de señas mexicana, capturando todo el encuadre de la persona en video, al menos desde la cintura hasta la cabeza.
- El enfoque actual supera la viabilidad de realizar investigaciones previas utilizando bases de datos con solo manos recortadas y el uso de sensores complejos relacionados con LSM, como lo demuestran las grabaciones de escenarios del mundo real del estudio actual con algoritmos de última generación aplicados de la literatura.
- Se desarrolló un proyecto de investigación que apoya la comunicación para las comunidades sordas, beneficiando así a la nación y a los objetivos sostenibles de la ONU (Objetivos 4, 8, 9, 10 y 11).

Referencias

- Adeyanju, I. A., Bello, O. O., & Adegboye, M. A. (2021). *Machine learning methods for sign language recognition: A critical review and analysis. Intelligent Systems with Applications, 12*, 200056. <https://doi.org/10.1016/j.iswa.2021.200056>
- Alzamzami, F., Hoda, M., & El Saddik, A. (2020). *Light Gradient Boosting Machine for General Sentiment Classification on Short Texts: A Comparative Evaluation. IEEE Access, 8*, 101840–101858. <https://doi.org/10.1109/ACCESS.2020.2997330>
- Ardiansyah, A., Hitoyoshi, B., Halim, M., Hanafiah, N., & Wibisurya, A. (2021). *Systematic Literature Review: American Sign Language Translator. Procedia Computer Science, 179*, 541–549. <https://doi.org/10.1016/j.procs.2021.01.038>
- Ahmed, M. A., Zaidan, B. B., Zaidan, A. A., Salih, M. M., & Lakulu, M. M. bin. (2018). *A Review on Systems-Based Sensory Gloves for Sign Language Recognition State of the Art between 2007 and 2017. Sensors, 18(7)*, 2208. <https://doi.org/10.3390/s18072208>
- Bantupalli, K., & Xie, Y. (2019). *American Sign Language Recognition Using Machine Learning and Computer Vision [Kennesaw State University]*. https://digitalcommons.kennesaw.edu/cs_etd/21
- Bilgin, M., & Mutludogan, K. (2019). *American Sign Language Character Recognition with Capsule Networks. 2019 3rd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), 1–6*. <https://doi.org/10.1109/ISMSIT.2019.8932829>
- BrightSign Technology Limited. (2023). *The BrightSign Glove*. <https://www.brightsignglove.com/>
- Camgoz, N. C., Koller, O., Hadfield, S., & Bowden, R. (2020). *Sign Language Transformers:*

- Joint End-to-end Sign Language Recognition and Translation* (arXiv:2003.13830).
arXiv. <http://arxiv.org/abs/2003.13830>
- Chen, R.-C., Manongga, W. E., & Dewi, C. (2022). *Recursive Feature Elimination for Improving Learning Points on Hand-Sign Recognition*. *Future Internet*, 14(12), 352.
<https://doi.org/10.3390/fi14120352>
- Consejo para Prevenir y Eliminar la Discriminación en la Ciudad de México (COPRED). (2019). *Lenguas de señas son fundamentales para el desarrollo de las personas sordas y el acceso a sus derechos*.
<https://copred.cdmx.gob.mx/comunicacion/nota/lenguas-de-senas-son-fundamentales-para-el-desarrollo-de-las-personas-sordas-y-el-acceso-sus-derechos>
- Department of Economic and Social Affairs of UN. (2019, Septiembre 17). *International Day of Sign Languages, 23 September 2019*.
<https://www.un.org/development/desa/disabilities/news/news/sign-languages.html>
- Elhagry, A., & Elrayes, R. G. (2021). *Egyptian Sign Language Recognition Using CNN and LSTM* (arXiv:2107.13647). arXiv. <http://arxiv.org/abs/2107.13647>
- EMEET. (2023, Febrero 23). *WebCam 1080p*. https://www.amazon.com/-/en/C3%A1mara-web-1080P-privacidad-cancelaci%C3%B3n/dp/B09KCJ36FS/ref=sr_1_4?keywords=webcam&qid=1677513484&sr=8-4
- Espejel-Cabrera, J., Cervantes, J., García-Lamont, F., Ruiz Castilla, J. S., & D. Jalili, L. (2021). *Mexican sign language segmentation using color based neuronal networks to detect the individual skin color*. *Expert Systems with Applications*, 183, 115295.
<https://doi.org/10.1016/j.eswa.2021.115295>
- Estados Unidos Mexicanos. (2008). *Ley General de las Personas con Discapacidad*. General

Secretary; DOF 30-05-2011.

https://www.diputados.gob.mx/LeyesBiblio/abro/lgpd/LGPD_abro.pdf

Geng, J., Huang, D., & De la Torre, F. (2023). *DensePose From WiFi*.

<https://doi.org/10.48550/ARXIV.2301.00250>

Géron, A. (2019). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems* (Second edition).

O'Reilly Media, Inc.

Grandini, M., Bagli, E., & Visani, G. (2020). *Metrics for Multi-Class Classification: An Overview*. <https://doi.org/10.48550/ARXIV.2008.05756>

Güler, R. A., Neverova, N., & Kokkinos, I. (2018). *DensePose: Dense Human Pose Estimation In The Wild*. <https://doi.org/10.48550/ARXIV.1802.00434>

Guzsvinecz, T., Szucs, V., & Sik-Lanyi, C. (2019). *Suitability of the Kinect Sensor and Leap Motion Controller—A Literature Review*. *Sensors*, 19(5), 1072.

<https://doi.org/10.3390/s19051072>

Halder, A., & Tayade, A. (2021). *Real-time Vernacular Sign Language Recognition using MediaPipe and Machine Learning*. *International Journal of Research Publication and Reviews*, 2(5), 9–17.

Indriani, Harris, Moh., & Agoes, A. S. (2021). *Applying Hand Gesture Recognition for User Guide Application Using MediaPipe*: 2nd International Seminar of Science and

Applied Technology (ISSAT 2021), Bandung, Indonesia.

<https://doi.org/10.2991/aer.k.211106.017>

Jimenez, J., Martin, A., Uc, V., & Espinosa, A. (2017). *Mexican Sign Language Alphanumeric Gestures Recognition using 3D Haar-like Features*. *IEEE Latin*

America Transactions, 15(10), 2000–2005.

<https://doi.org/10.1109/TLA.2017.8071247>

Kim, E. J., Byrne, B., & Parish, S. L. (2018). *Deaf people and economic well-being: Findings from the Life Opportunities Survey*. *Disability & Society*, 33(3), 374–391.

<https://doi.org/10.1080/09687599.2017.1420631>

Koishybay, K., Mukushev, M., & Sandygulova, A. (2021). *Continuous Sign Language Recognition with Iterative Spatiotemporal Fine-tuning*. *2020 25th International Conference on Pattern Recognition (ICPR)*, 10211–10218.

<https://doi.org/10.1109/ICPR48806.2021.9412364>

Li, D., Opazo, C. R., Yu, X., & Li, H. (2020). *Word-level Deep Sign Language Recognition from Video: A New Large-scale Dataset and Methods Comparison* (arXiv:1910.11006). arXiv. <http://arxiv.org/abs/1910.11006>

Liu, P., Li, X., Cui, H., Li, S., & Yuan, Y. (2019). *Hand Gesture Recognition Based on Single-Shot Multibox Detector Deep Learning*. *Mobile Information Systems, 2019*, 1–7. <https://doi.org/10.1155/2019/3410348>

Lum, K. Y., Goh, Y. H., & Lee, Y. B. (2020). *American Sign Language Recognition Based on MobileNetV2*. *Advances in Science, Technology and Engineering Systems Journal*, 5(6), 481–488. <https://doi.org/10.25046/aj050657>

Mariano, M. (2020, Agosto 14). *Enfrentan personas con discapacidad auditiva retos en escenario actual*. <https://conecta.tec.mx/es/noticias/monterrey/educacion/enfrentan-personas-con-discapacidad-auditiva-retos-en-escenario-actual>

MathWorks (2023). Support Vector Machine (SVM). Hiperplanos óptimos como límites de decisión. Consultado el 12 de julio de 2023. Obtenido de https://la.mathworks.com/discovery/support-vector-machine.html

Mejía-Peréz, K., Córdova-Esparza, D.-M., Terven, J., Herrera-Navarro, A.-M., García-

- Ramírez, T., & Ramírez-Pedraza, A. (2022). *Automatic Recognition of Mexican Sign Language Using a Depth Camera and Recurrent Neural Networks*. *Applied Sciences*, 12(11), 5523. <https://doi.org/10.3390/app12115523>
- Min, Y., Hao, A., Chai, X., & Chen, X. (2021). *Visual Alignment Constraint for Continuous Sign Language Recognition*. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 11522–11531. <https://doi.org/10.1109/ICCV48922.2021.01134>
- Mohammedali, A. H., Abbas, H. H., & Shahadi, H. I. (2022). *Real-time sign language recognition system*. *International Journal of Health Sciences*, 10384–10407. <https://doi.org/10.53730/ijhs.v6nS4.12206>
- Mujahid, A., Awan, M. J., Yasin, A., Mohammed, M. A., Damaševičius, R., Maskeliūnas, R., & Abdulkareem, K. H. (2021). *Real-Time Hand Gesture Recognition Based on Deep Learning YOLOv3 Model*. *Applied Sciences*, 11(9), 4164. <https://doi.org/10.3390/app11094164>
- Rathi, D. (2018). *Optimization of Transfer Learning for Sign Language Recognition Targeting Mobile Platform* (arXiv:1805.06618). arXiv. <http://arxiv.org/abs/1805.06618>
- Renotte, N. (2020). *ActionDetectionforSignLanguage*. *GitHub*. <https://github.com/nicknochnack/ActionDetectionforSignLanguage>
- Rios-Figueroa, H. V., Sánchez-García, A. J., Sosa-Jiménez, C. O., & Solís-González-Cosío, A. L. (2022). *Use of Spherical and Cartesian Features for Learning and Recognition of the Static Mexican Sign Language Alphabet*. *Mathematics*, 10(16), 2904. <https://doi.org/10.3390/math10162904>
- Rodrigo, J. (2020). *Máquinas de Vector Soporte (SVM) con Python*. Consultado el 12 de julio de 2023. Obtenido de <https://cienciadedatos.net/documentos/py24-svm-python.html>

- Rodriguez, M., Outmane, O., Bassam, A., & Noureddine, L. (2023a). *Mexican Sign Language's Dactylology and Ten First Numbers -Raw videos. From person #1 to #3* [Data set]. Mendeley. V1. <https://doi.org/10.17632/48XYBSMVPV.1>
- Rodriguez, M., Outmane, O., Bassam, A., & Noureddine, L. (2023b). *Mexican Sign Language's Dactylology and Ten First Numbers -Raw videos. From person #4 to #7* [Data set]. Mendeley. V1. <https://doi.org/10.17632/JP4YMF2VJW.1>
- Rodriguez, M., Outmane, O., Bassam, A., & Noureddine, L. (2023c). *Mexican Sign Language's Dactylology and Ten First Numbers -Raw videos. From person #8 to #10* [Data set]. Mendeley. V1. <https://doi.org/10.17632/69FMDB25XM.1>
- Rodriguez, M., Outmane, O., Bassam, A., & Noureddine, L. (2023d). *Mexican Sign Language's Dactylology and Ten First Numbers -Raw videos. Person #11 [Data set]*. Mendeley. V1. <https://doi.org/10.17632/3DZN5RSTWX.1>
- Rodriguez, M., Outmane, O., Bassam, A., & Noureddine, L. (2023e). *Mexican Sign Language's Dactylology and Ten First Numbers—Extracted Features and Models [Data set]*. Mendeley. V1. <https://doi.org/10.17632/HMSC33HMKZ.1>
- Rodriguez, M., Outmane, O., Bassam, A. & Noureddine, L. (2023f). *Mexican Sign Language's Dactylology and Ten First Numbers—Labeled images and videos. From person #1 to #5 [Data set]*. Mendeley. V1. <https://doi.org/10.17632/5S4MT7XRD9.1>
- Rodriguez, M., Outmane, O., Bassam, A., & Noureddine, L. (2023g). *Mexican Sign Language's Dactylology and Ten First Numbers—Labeled images and videos. From person #6 to #11 [Data set]*. Mendeley. V1. <https://doi.org/10.17632/67HTNZMWBB.1>
- Rodriguez, M., Outmane, O., Bassam, A., & Noureddine, L. (2023h).

TrainingAndExtractionForMSLRecognition. Github.

<https://github.com/EmmanuelRTM/TrainingAndExtractionForMSLRecognition>

Samaan, G. H., Wadie, A. R., Attia, A. K., Asaad, A. M., Kamel, A. E., Slim, S. O., Abdallah, M. S., & Cho, Y.-I. (2022). *MediaPipe's Landmarks with RNN for Dynamic Sign Language Recognition*. *Electronics*, 11(19), 3228. <https://doi.org/10.3390/electronics11193228>

Scikit-Learn (2023). *Ensembled Models. Histogram-Based Gradient Boosting*. Consultado el 15 de julio de 2023. Obtenido de <https://scikit-learn.org/stable/modules/ensemble.html#histogram-based-gradient-boosting>

Serafin , M. E., & González, R. (2011). *Manos con voz. Diccionario de la Lengua de Señas Mexicana*. Consejo Nacional para Prevenir la Discriminación (CONAPRED) & Libre Acceso, A.C. Primera Edición. ISBN 978-607-7514-35-0 https://www.conapred.org.mx/documentos_cedoc/DiccioSenas_ManosVoz_ACCSS.pdf

Shin, J., Matsuoka, A., Hasan, Md. A. M., & Srizon, A. Y. (2021). *American Sign Language Alphabet Recognition by Extracting Feature from Hand Pose Estimation*. *Sensors*, 21(17), 5856. <https://doi.org/10.3390/s21175856>

Solís, F., Martínez, D., & Espinoza, O. (2016). *Automatic Mexican Sign Language Recognition Using Normalized Moments and Artificial Neural Networks*. *Engineering*, 08(10), 733–740. <https://doi.org/10.4236/eng.2016.810066>

Solís, F., Toxqui, C., & Martínez, D. (2015). *Mexican Sign Language Recognition Using Jacobi-Fourier Moments*. *Engineering*, 07(10), 700–705. <https://doi.org/10.4236/eng.2015.710061>

Sosa-Jimenez, C. O., Rios-Figueroa, H. V., & Solis-Gonzalez-Cosio, A. L. (2022). A

- Prototype for Mexican Sign Language Recognition and Synthesis in Support of a Primary Care Physician. IEEE Access, 10, 127620–127635. <https://doi.org/10.1109/ACCESS.2022.3226696>*
- Subramanian, B., Olimov, B., Naik, S. M., Kim, S., Park, K.-H., & Kim, J. (2022). *An integrated mediapipe-optimized GRU model for Indian sign language recognition. Scientific Reports, 12(1), 11964. <https://doi.org/10.1038/s41598-022-15998-7>*
- Sundar, B., & Bagyammal, T. (2022). *American Sign Language Recognition for Alphabets Using MediaPipe and LSTM. Procedia Computer Science, 215, 642–651. <https://doi.org/10.1016/j.procs.2022.12.066>*
- TheAIGuysCode. (2020). *Colab-webcam. GitHub. <https://github.com/theAIGuysCode/colab-webcam>*
- UltraLeap. (2023, Febrero 23). *Leap Motion Controller. <https://www.ultraleap.com/product/leap-motion-controller/>*
- United Nations. (2016). *The 17 Goals. <https://sdgs.un.org/goals>*
- World Health Organization. (2023, Febrero 27). *Deafness and hearing loss. <https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss>*
- Xbox One. (2023, Febrero 23). *Xbox One Kinect Sensor. https://www.amazon.com/-/en/Xbox-One-Kinect-Sensor/dp/B00INAX3Q2/ref=sr_1_7?keywords=Kinect+for+Xbox+One&qid=1677511501&sr=8-7*
- Zhang, F., Bazarevsky, V., Vakunov, A., Tkachenka, A., Sung, G., Chang, C.-L., & Grundmann, M. (2020). *MediaPipe Hands: On-device Real-time Hand Tracking (arXiv:2006.10214). arXiv. <http://arxiv.org/abs/2006.10214>*

Apéndice

Apéndice 1. Arquitectura de Modelos LSTM y GRU

Layer (type)	Output Shape	Param #
lstm_3 (LSTM)	(None, 30, 64)	48896
lstm_4 (LSTM)	(None, 30, 128)	98816
lstm_5 (LSTM)	(None, 64)	49408
dense_6 (Dense)	(None, 64)	4160
dense_7 (Dense)	(None, 32)	2080
dense_8 (Dense)	(None, 10)	330

=====
Total params: 203,690
Trainable params: 203,690
Non-trainable params: 0

Figura 20. Arquitectura del modelo LSTM.

Layer (type)	Output Shape	Param #
gru (GRU)	(None, 30, 128)	98304
dropout (Dropout)	(None, 30, 128)	0
dense_2 (Dense)	(None, 30, 64)	8256
gru_1 (GRU)	(None, 30, 64)	24960
dropout_1 (Dropout)	(None, 30, 64)	0
dense_3 (Dense)	(None, 30, 64)	4160
gru_2 (GRU)	(None, 32)	9408
batch_normalization (Batch Normalization)	(None, 32)	128
dense_4 (Dense)	(None, 32)	1056
dense_5 (Dense)	(None, 10)	330

=====
Total params: 146,602
Trainable params: 146,538
Non-trainable params: 64

Figura 21. Arquitectura del modelo GRU.

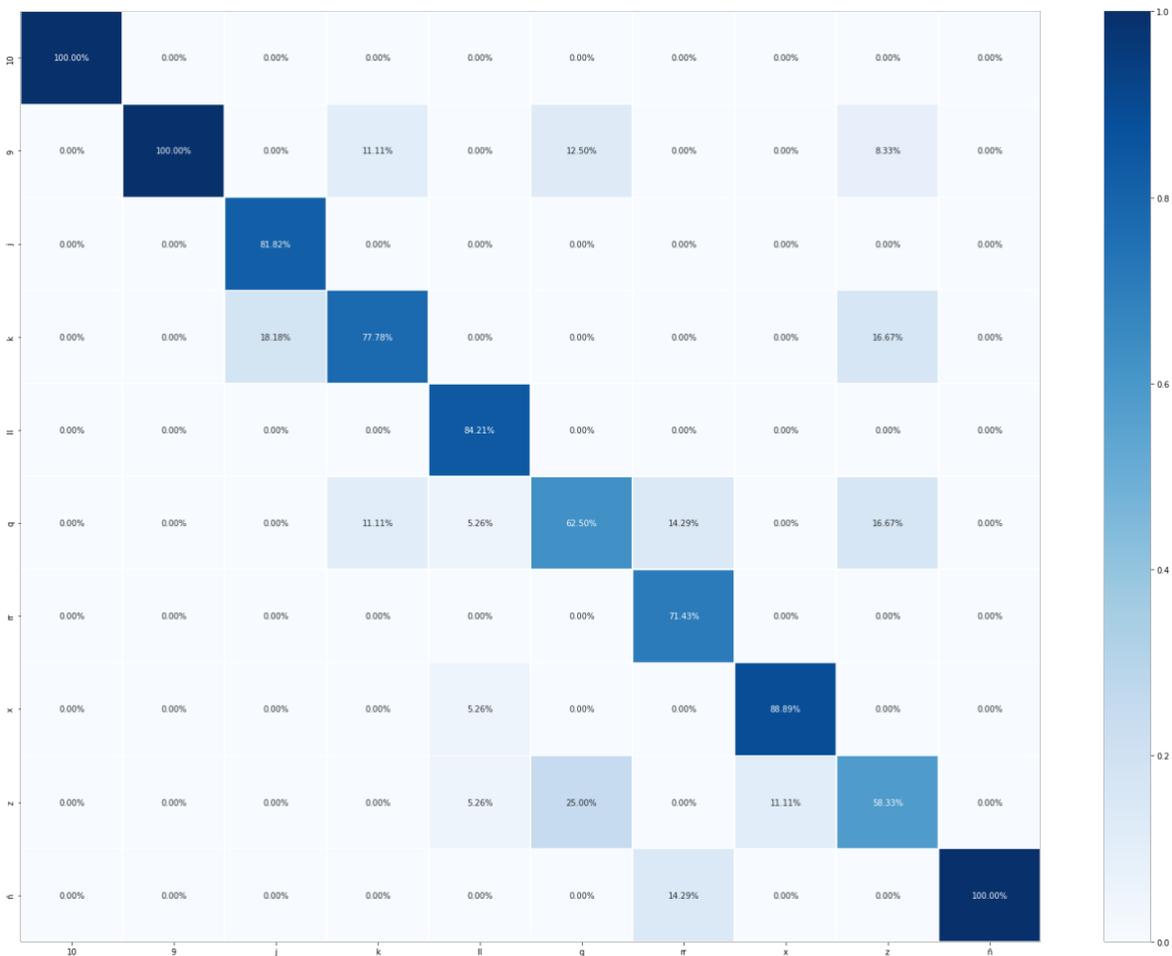


Figura 24. Matriz de Confusión de las Señas Continuas del LSM con el modelo LSTM.

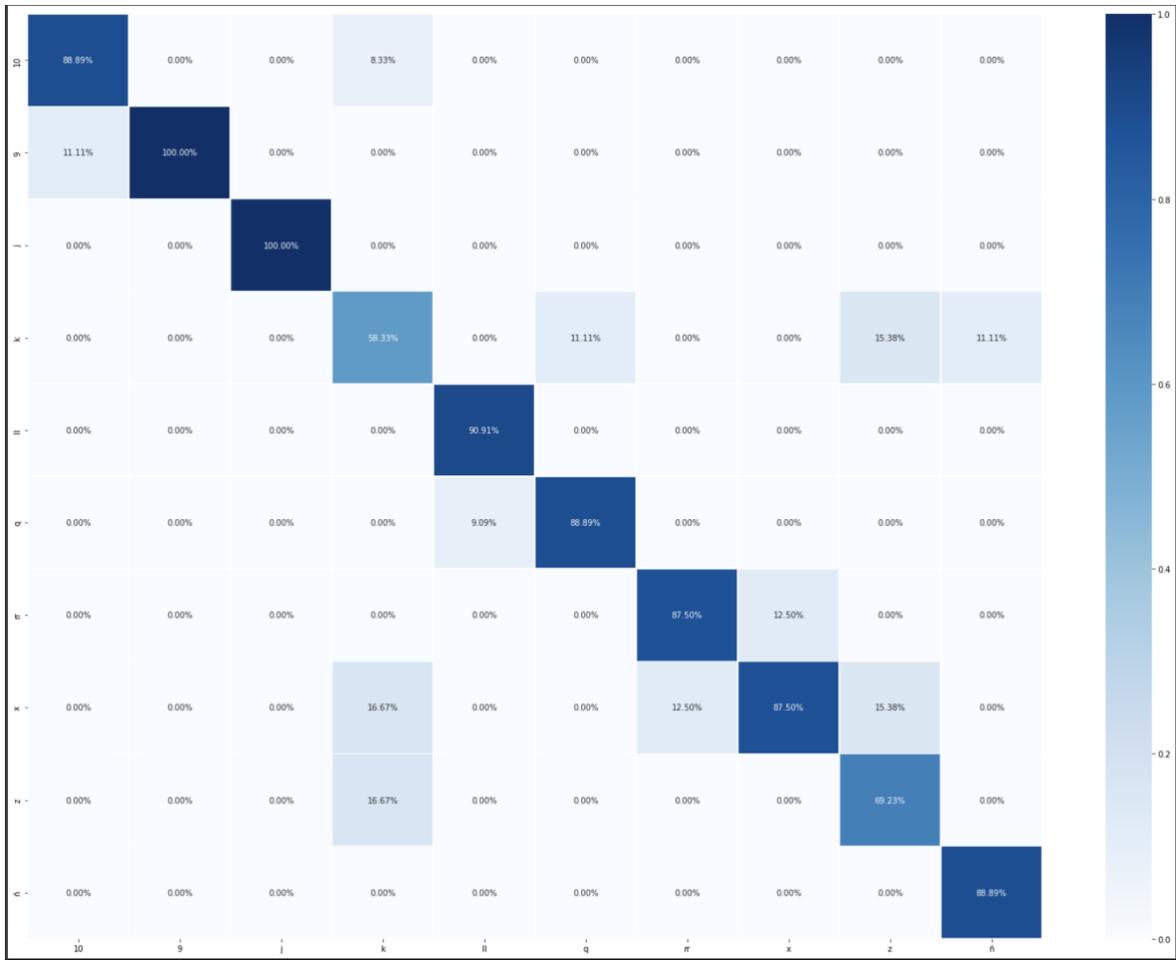


Figura 25. Matriz de Confusión de las Señas Continuas del LSM con el modelo GRU.

Lista de Figuras

- Figura 1. Número de publicaciones de artículos en relación a la lengua de señas de Adeyanjua, I.A., Bellob,O.O., & Adegboyea, M.A., (2021).
- Figura 2. Países con mayor número de publicaciones de artículos en relación a la lengua de señas de Adeyanjua, I.A., Bellob,O.O., & Adegboyea, M.A., (2021).
- Figura 3. Colaboraciones entre países de publicaciones de artículos en relación a la lengua de señas de Adeyanjua, I.A., Bellob,O.O., & Adegboyea, M.A., (2021).
- Figura 4. Diagrama de métodos de preprocesamiento de imágenes de Adeyanjua, I.A., Bellob,O.O., & Adegboyea, M.A., (2021).
- Figura 5. Diagrama de técnicas de segmentación de imágenes de Adeyanjua, I.A., Bellob,O.O., & Adegboyea, M.A., (2021).
- Figura 6. Diagrama del funcionamiento del algoritmo de *MediaPipe* (BlazePalm Detector) de Zhang et al. (2020).
- Figuras 7 y 8. Marcas de las posiciones de *MediaPipe* en sus 21 puntos de las manos de Zahng et al. (2020).
- Figura 9. Muestra del uso de *MediaPipe Holistic* para la detección de manos.
- Figura 10. Muestra de definición del “margen” entre clases: el criterio que los SVM intentan optimizar de *MathWorks* (2023).
- Figura 11. Celda LSTM de Géron, A. (2019).
- Figura 12. Celda GRU de Géron, A. (2019).
- Figura 13. Propuesta de metodología de investigación. Elaboración propia, basada en Sampieri et al. (1997) y Ardiansyah, A., et al. (2021)
- Figura 14. Proceso adaptado para el reconocimiento de lengua de señas.
- Figura 15. Ejemplo de toma de la señal Número 2 (Señal Estática).
- Figura 16. Arquitectura de GRU y LSTM
- Figura 17. Gráficos de pérdida de época y exactitud categórica de LSTM.
- Figura 18. Gráficos de pérdida de época y exactitud categórica de GRU.
- Figura 19. Ejemplo de la letra “V” Reconocida.
- Figura 20. Arquitectura del modelo LSTM.
- Figura 21. Arquitectura del modelo GRU.
- Figura 22. Matriz de Confusión de las Señas Estáticas del LSM con el modelo SVM.
- Figura 23. Matriz de Confusión de las Señas Estáticas del LSM con el modelo GBL.
- Figura 24. Matriz de Confusión de las Señas Continuas del LSM con el modelo LSTM.
- Figura 25. Matriz de Confusión de las Señas Continuas del LSM con el modelo GRU.

Lista de Tablas

- Tabla 1. Tabla de los Métodos de obtención de imágenes de Ardiansyah, A., et al. (2021)
- Tabla 2. Tabla de los Métodos de preprocesamiento de imágenes de Ardiansyah, A., et al. (2021)
- Tabla 3. Tabla de los Métodos de segmentación de Ardiansyah, A., et al. (2021)
- Tabla 4. Tabla de los Métodos de los métodos de extracción de características de Ardiansyah, A., et al. (2021)
- Tabla 5. Tabla de los Métodos de clasificación de Ardiansyah, A., et al. (2021)
- Tabla 6. Tabla de los Métodos de clasificación de ML de Adeyanjua, I.A., Bellob,O.O., & Adegboyea, M.A., (2021).
- Tabla 7. Modelos aplicados para el Reconocimiento de LS con CNN y RNN.
- Tabla 8. Tabla Comparativa de la Revisión de Literatura aplicada al Reconocimiento de Lengua de Señas relacionada con *MediaPipe* y sin el.
- Tabla 9. Conjunto de datos etiquetados y el número de muestras para las señas estáticas.
- Tabla 10. Conjunto de datos etiquetados y el número de muestras para las señas continuas.
- Tabla 11. Parámetros para señas del LSM estáticas.
- Tabla 12. Métricas de evaluación.

“Por una Facultad innovadora creando un futuro sostenible”

Cuernavaca, Morelos a 31 de Agosto del 2023.

DR. FELIPE DE JESÚS BONILLA SÁNCHEZ
DIRECTOR DE LA FACULTAD DE CONTADURÍA,
ADMINISTRACIÓN E INFORMÁTICA DE LA UAEM.
PRESENTEPRESENTE

En mi carácter de revisor de Tesis, hago de su conocimiento que he leído con interés la tesis para obtener el grado de la Maestría en Optimización y Cómputo Aplicado, del estudiante **Mario Emmanuel Rodríguez Trejo**, con matrícula **10053358**, con el título **"Modelos Computacionales Aplicados a la Lengua de Señas Mexicana"**, por lo cual, me permito informarle que después de una revisión cuidadosa de dicha tesis, concluyo que el trabajo se caracteriza por el establecimiento de objetivos académicos pertinentes y una metodología adecuada para su logro. Además, construye una estructura coherente y bien documentada, por lo cual considero que los resultados obtenidos contribuyen al conocimiento del tema tratado.

Con base en los argumentos precedentes me permito expresar mi **VOTO APROBATORIO** por lo que de mi parte no existe inconveniente para que el estudiante continúe con los trámites que esta Secretaría de Investigación tenga establecidos para obtener el grado mencionado.

Atentamente
Por una humanidad culta
Una universidad de excelencia

Dr. Outmane Oubram
Profesor- investigador
Facultad de Ciencias Químicas e Ingeniería





UNIVERSIDAD AUTÓNOMA DEL
ESTADO DE MORELOS

Se expide el presente documento firmado electrónicamente de conformidad con el ACUERDO GENERAL PARA LA CONTINUIDAD DEL FUNCIONAMIENTO DE LA UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS DURANTE LA EMERGENCIA SANITARIA PROVOCADA POR EL VIRUS SARS-COV2 (COVID-19) emitido el 27 de abril del 2020.

El presente documento cuenta con la firma electrónica UAEM del funcionario universitario competente, amparada por un certificado vigente a la fecha de su elaboración y es válido de conformidad con los LINEAMIENTOS EN MATERIA DE FIRMA ELECTRÓNICA PARA LA UNIVERSIDAD AUTÓNOMA DE ESTADO DE MORELOS emitidos el 13 de noviembre del 2019 mediante circular No. 32.

Sello electrónico

OUTMANE OUBRAM | Fecha:2023-08-31 10:11:24 | Firmante

KZyo/W1ttKm9W0e5KDnpiLVI32WR2aHwD91D720t5qfz741K/kH5u9dW0xE3aQ0vcWsZFmxlgZd/wuq0rGlmDgfiK1e6Y5zrQwmRo10qr0c+6ZHKyT70amcEr+inBzff/mlbtPpeHA0S2TbvqfaTiH+P7KS4kgaB5hD6p0CVwNZzdsd1g4L8Hdn9H6iDR/7iawWGYvDXxu2qLVfVfCf4Ww92JimXy8Q5/BuKppMPdCF4owPqjf7nGE7BinCamtCj5tnADIEGF2ba9rsAV9VfQwp7wJJ42vpCb/wLQH8OoUL7bwnK0wjlxTOCpoKeMIDZAUG566QkP3C9o7QQCh7w==

Puede verificar la autenticidad del documento en la siguiente dirección electrónica o escaneando el código QR ingresando la siguiente clave:



[3i7dHqgMh](#)

<https://efirma.uaem.mx/noRepudio/IHZ8bWXCRCDD0r7zRt5nJ3z369lvPmWRh>



Cuernavaca, Morelos a 31 de Agosto del 2023.

DR. FELIPE DE JESÚS BONILLA SÁNCHEZ
DIRECTOR DE LA FACULTAD DE CONTADURÍA,
ADMINISTRACIÓN E INFORMÁTICA DE LA UAEM.
PRESENTEPRESENTE

En mi carácter de revisor de Tesis, hago de su conocimiento que he leído con interés la tesis para obtener el grado de la Maestría en Optimización y Cómputo Aplicado, del estudiante **Mario Emmanuel Rodríguez Trejo**, con matrícula **10053358**, con el título **"Modelos Computacionales Aplicados a la Lengua de Señas Mexicana"**, por lo cual, me permito informarle que después de una revisión cuidadosa de dicha tesis, concluyo que el trabajo se caracteriza por el establecimiento de objetivos académicos pertinentes y una metodología adecuada para su logro. Además, construye una estructura coherente y bien documentada, por lo cual considero que los resultados obtenidos contribuyen al conocimiento del tema tratado.

Con base en los argumentos precedentes me permito expresar mi **VOTO APROBATORIO** por lo que de mi parte no existe inconveniente para que el estudiante continúe con los trámites que esta Secretaría de Investigación tenga establecidos para obtener el grado mencionado.

Atentamente
Por una humanidad culta
Una universidad de excelencia

Dr, Nouredine Lakouari
Profesor- investigador
Instituto nacional de astrofísica óptica y electrónica (INAOE)





UNIVERSIDAD AUTÓNOMA DEL
ESTADO DE MORELOS

Se expide el presente documento firmado electrónicamente de conformidad con el ACUERDO GENERAL PARA LA CONTINUIDAD DEL FUNCIONAMIENTO DE LA UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS DURANTE LA EMERGENCIA SANITARIA PROVOCADA POR EL VIRUS SARS-COV2 (COVID-19) emitido el 27 de abril del 2020.

El presente documento cuenta con la firma electrónica UAEM del funcionario universitario competente, amparada por un certificado vigente a la fecha de su elaboración y es válido de conformidad con los LINEAMIENTOS EN MATERIA DE FIRMA ELECTRÓNICA PARA LA UNIVERSIDAD AUTÓNOMA DE ESTADO DE MORELOS emitidos el 13 de noviembre del 2019 mediante circular No. 32.

Sello electrónico

NOUREDDINE LAKOUARI | Fecha:2023-08-31 11:51:46 | Firmante

xeEzswIXRPInRM47AHAKMonuddA5t6kpv5dJUdCZocGLsoL7WOOzhSpWOkdic+geRWwceinLSOPnCWAqa/mP5O0ywbmCOHBZeSEnsFtNQFNrUR6NQz+IruNNIOmPuecVyLUjba5dCtas1uO+smi8wawzll+RdS5kFeLUbSZ8WfRnTAaqfJet0tkODM/OY1qTjpBv+1hIPzFuknJKcbwuFc7Md5qO2vLdy0aCSL0Mqj5vdpIifNmMRIfSKDvW4+Y2izmu6tFcrRXO/9CL+j3kQ1maSTihZV4evwnz76N6a9y8FOW1b1EpPi1oxdNareDIUZelE5vb58PLxWdhu774bg==

Puede verificar la autenticidad del documento en la siguiente dirección electrónica o escaneando el código QR ingresando la siguiente clave:



[EQLmvfkDb](#)

<https://efirma.uaem.mx/noRepudio/ZA3Wg5A43kx4NLRXGrCk3R5INwFzDyyd>



Cuernavaca, Morelos a 31 de Agosto del 2023.

DR. FELIPE DE JESÚS BONILLA SÁNCHEZ
DIRECTOR DE LA FACULTAD DE CONTADURÍA,
ADMINISTRACIÓN E INFORMÁTICA DE LA UAEM.
PRESENTEPRESENTE

En mi carácter de revisor de Tesis, hago de su conocimiento que he leído con interés la tesis para obtener el grado de la Maestría en Optimización y Cómputo Aplicado, del estudiante **Mario Emmanuel Rodríguez Trejo**, con matrícula **10053358**, con el título **"Modelos Computacionales Aplicados a la Lengua de Señas Mexicana"**, por lo cual, me permito informarle que después de una revisión cuidadosa de dicha tesis, concluyo que el trabajo se caracteriza por el establecimiento de objetivos académicos pertinentes y una metodología adecuada para su logro. Además, construye una estructura coherente y bien documentada, por lo cual considero que los resultados obtenidos contribuyen al conocimiento del tema tratado.

Con base en los argumentos precedentes me permito expresar mi **VOTO APROBATORIO** por lo que de mi parte no existe inconveniente para que el estudiante continúe con los trámites que esta Secretaría de Investigación tenga establecidos para obtener el grado mencionado.

Atentamente
Por una humanidad culta
Una universidad de excelencia

Dr. Ali Bassam
Profesor - investigador y Coordinador C.A. Energías Renovables y Sustentabilidad Energética
Facultad de Ingeniería de la Universidad Autónoma de Yucatán (UADY)





UNIVERSIDAD AUTÓNOMA DEL
ESTADO DE MORELOS

Se expide el presente documento firmado electrónicamente de conformidad con el ACUERDO GENERAL PARA LA CONTINUIDAD DEL FUNCIONAMIENTO DE LA UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS DURANTE LA EMERGENCIA SANITARIA PROVOCADA POR EL VIRUS SARS-COV2 (COVID-19) emitido el 27 de abril del 2020.

El presente documento cuenta con la firma electrónica UAEM del funcionario universitario competente, amparada por un certificado vigente a la fecha de su elaboración y es válido de conformidad con los LINEAMIENTOS EN MATERIA DE FIRMA ELECTRÓNICA PARA LA UNIVERSIDAD AUTÓNOMA DE ESTADO DE MORELOS emitidos el 13 de noviembre del 2019 mediante circular No. 32.

Sello electrónico

BASSAM ALI | Fecha:2023-09-04 00:20:52 | Firmante

c5hdzW1W9BEAvfV6CpcMYVRvfVGUL7A4IThxfvPH5umgYtQM4irKDgCKWdutph5Htpx5X5FH7YbkEyp2ccg8IU1ON9uMmBuFFndG7YUw5srUWXOBrlu8JAqSnW4ZryPg7f5
lrmfXfz0FQvDOOKr2hH1VMftrm7dwo5g9Vy1zht8llHHxLceKcxDI+pud1wBYRSBRE+c1DGZhxGLVqpPhHi0IEhMTL5XdCNzk8nt3z8YcPtQmmnteivDznvz+WGNDAJGjR2LNg
Clo479oW7AtGL4eSt4bX7lhhYi2q+xoHGU8K4ZJSJXCZ1akBu//9LMUKfLkmlCjCNtDYyUw==

Puede verificar la autenticidad del documento en la siguiente dirección electrónica o
escaneando el código QR ingresando la siguiente clave:



[W4u5wokdE](#)

<https://efirma.uaem.mx/noRepudio/zTLaFwhZk1h3Dutolrs8fwEA7OnzG8G2>



Cuernavaca, Morelos a 4 de septiembre del 2023.

DR. FELIPE DE JESÚS BONILLA SÁNCHEZ
DIRECTOR DE LA FACULTAD DE CONTADURÍA,
ADMINISTRACIÓN E INFORMÁTICA DE LA UAEM.
PRESENTEPRESENTE

En mi carácter de revisor de Tesis, hago de su conocimiento que he leído con interés la tesis para obtener el grado de la Maestría en Optimización y Cómputo Aplicado, del estudiante **Mario Emmanuel Rodríguez Trejo**, con matrícula **10053358**, con el título **"Modelos Computacionales Aplicados a la Lengua de Señas Mexicana"**, por lo cual, me permito informarle que después de una revisión cuidadosa de dicha tesis, concluyo que el trabajo se caracteriza por el establecimiento de objetivos académicos pertinentes y una metodología adecuada para su logro. Además, construye una estructura coherente y bien documentada, por lo cual considero que los resultados obtenidos contribuyen al conocimiento del tema tratado.

Con base en los argumentos precedentes me permito expresar mi **VOTO APROBATORIO** por lo que de mi parte no existe inconveniente para que el estudiante continúe con los trámites que esta Secretaría de Investigación tenga establecidos para obtener el grado mencionado.

Atentamente
Por una humanidad culta
Una universidad de excelencia

Dra. Lorena Díaz González
Profesora- investigadora
Centro de Investigación en Ciencias





UNIVERSIDAD AUTÓNOMA DEL
ESTADO DE MORELOS

Se expide el presente documento firmado electrónicamente de conformidad con el ACUERDO GENERAL PARA LA CONTINUIDAD DEL FUNCIONAMIENTO DE LA UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS DURANTE LA EMERGENCIA SANITARIA PROVOCADA POR EL VIRUS SARS-COV2 (COVID-19) emitido el 27 de abril del 2020.

El presente documento cuenta con la firma electrónica UAEM del funcionario universitario competente, amparada por un certificado vigente a la fecha de su elaboración y es válido de conformidad con los LINEAMIENTOS EN MATERIA DE FIRMA ELECTRÓNICA PARA LA UNIVERSIDAD AUTÓNOMA DE ESTADO DE MORELOS emitidos el 13 de noviembre del 2019 mediante circular No. 32.

Sello electrónico

LORENA DIAZ GONZALEZ | Fecha:2023-09-04 09:37:40 | Firmante

wPGKf8l7a+8bdIRnpWfNenFL941w7XuQOAPcNfPfJOflfhQ8iCIHzxq4Sibi9jBEQXTleKBCB8UW1fx1ccLLRPwhlf7P4VPD/MRldkpVZonYbQWdgm6fBCT1x4hUalYmnlwbV30lc6o
gU0ogdfkOi7AKATZXwuXa6ZICYrqU8nqWO7R5l16pAZuXiaQggmS+fBzpf9cNWq/0+nLP7ZZLVaBT+0ioUO18P6Rq1fuSyFTM0m6nSauQ+Nl40sJ75OfwTShEsO6CWvCKvD7
u7oSnGUKYAJAc+e1i8mYz5BkzS4+h24fMMZpH9rsEqIWxwzeDGhfKLlJBukUzYjmBdfJw==

Puede verificar la autenticidad del documento en la siguiente dirección electrónica o
escaneando el código QR ingresando la siguiente clave:



[HEosl73j2](#)

<https://efirma.uaem.mx/noRepudio/R93Vzhwp6Q7p6BqdbT14sitaF6clOtfP>



Cuernavaca, Morelos a 5 de septiembre del 2023.

DR. FELIPE DE JESÚS BONILLA SÁNCHEZ
DIRECTOR DE LA FACULTAD DE CONTADURÍA,
ADMINISTRACIÓN E INFORMÁTICA DE LA UAEM.
PRESENTEPRESENTE

En mi carácter de revisor de Tesis, hago de su conocimiento que he leído con interés la tesis para obtener el grado de la Maestría en Optimización y Cómputo Aplicado, del estudiante **Mario Emmanuel Rodríguez Trejo**, con matrícula **10053358**, con el título **"Modelos Computacionales Aplicados a la Lengua de Señas Mexicana"**, por lo cual, me permito informarle que después de una revisión cuidadosa de dicha tesis, concluyo que el trabajo se caracteriza por el establecimiento de objetivos académicos pertinentes y una metodología adecuada para su logro. Además, construye una estructura coherente y bien documentada, por lo cual considero que los resultados obtenidos contribuyen al conocimiento del tema tratado.

Con base en los argumentos precedentes me permito expresar mi **VOTO APROBATORIO** por lo que de mi parte no existe inconveniente para que el estudiante continúe con los trámites que esta Secretaría de Investigación tenga establecidos para obtener el grado mencionado.

Atentamente
Por una humanidad culta
Una universidad de excelencia

Dr. Federico Alonso Pecina
Profesor- investigador
Facultad de Contaduría, Administración e Informática





UNIVERSIDAD AUTÓNOMA DEL
ESTADO DE MORELOS

Se expide el presente documento firmado electrónicamente de conformidad con el ACUERDO GENERAL PARA LA CONTINUIDAD DEL FUNCIONAMIENTO DE LA UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS DURANTE LA EMERGENCIA SANITARIA PROVOCADA POR EL VIRUS SARS-COV2 (COVID-19) emitido el 27 de abril del 2020.

El presente documento cuenta con la firma electrónica UAEM del funcionario universitario competente, amparada por un certificado vigente a la fecha de su elaboración y es válido de conformidad con los LINEAMIENTOS EN MATERIA DE FIRMA ELECTRÓNICA PARA LA UNIVERSIDAD AUTÓNOMA DE ESTADO DE MORELOS emitidos el 13 de noviembre del 2019 mediante circular No. 32.

Sello electrónico

FEDERICO ALONSO PECINA | Fecha:2023-09-05 15:54:11 | Firmante

ewjhz/pg3CcJY2KCZ9SeTuwa0PeIRQozteLCCzX5S3/kq5H8IPInj54262Tekc/MJ4YNbxqfBzbFvyVjy+x2Jxm52tKqsFoB5EX5KZy7D3ciz5zL/YAjwyZ+sho+LuByhU6155h7O9xye
mW03EheojkZezo0GjQCkS59Lcey/rh/aFaHxEpkq4+NoGU6wEYaxVVUbMU/6dt4/DOKTf50SKc5mJcX0gZZNNQ590sHgqdRZkd7PCA5YINGmL3F/HQxUN72gxRYqRurxiaW1
uqDXO4OVrWYbGSlwyG3wpPS3RYc1wp19xJTdkSYx6MI76UH37GcFZldx0i3C6AKJ1cPA==

Puede verificar la autenticidad del documento en la siguiente dirección electrónica o
escaneando el código QR ingresando la siguiente clave:



[HhnGEBRqy](#)

<https://efirma.uaem.mx/noRepudio/YydPn8qBaiBr749cd3jY7OTMbu6WVq6Y>

