

**UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MÉXICO**

**Facultad de Ingeniería**



**“RECONOCIMIENTO DE PATRONES DEL ALFABETO  
ESTÁTICO DE LA LENGUA DE SEÑAS MEXICANA CON  
APLICACIÓN DE LOS MOMENTOS INVARIANTES DE HU”**

**TESIS**

**QUE PARA OBTENER EL GRADO DE  
INGENIERO EN COMPUTACIÓN**

**PRESENTAN:**

**AREYCI HUERTA PATIÑO**

**YARA YESCENIA RIVAS SÁNCHEZ**

**ASESOR DE TESIS**

**M. en A. T. I. LUIS ENRIQUE LEDEZMA FUENTES**

**OCTUBRE 2023**

## ÍNDICE

<b>RESUMEN.....</b>	<b>3</b>
<b>INTRODUCCIÓN.....</b>	<b>4</b>
<b>Capítulo 1. PLANTEAMIENTO DEL PROBLEMA.....</b>	<b>6</b>
1.1 Antecedentes .....	6
1.2 Definición del Problema .....	9
1.3 Importancia del Tema .....	11
1.4 Pregunta de Investigación .....	11
1.5 Alcances del Problema .....	12
1.5.1 Contextual .....	12
1.5.2 Espacial .....	12
1.5.3 Temporalidad.....	12
1.5.4 Semántica o Disciplina .....	12
1.6 Objetivos .....	12
1.6.1 Objetivo General.....	12
1.6.2 Objetivos Particulares.....	13
1.7 Justificación.....	13
1.7.1 Institucional .....	14
1.7.2 Profesional .....	14
1.7.3 Personal.....	14
1.8 Hipótesis.....	15
1.9 Viabilidad.....	15
<b>CAPÍTULO 2. MARCO TEÓRICO Y ESTADO DEL ARTE .....</b>	<b>16</b>
2.1 Discapacidad Auditiva .....	16
2.2 Lengua de Señas Mexicana (LSM).....	18
2.3 Reconocimiento de Patrones (RP).....	21
2.4 Técnicas y Métodos Basados en el RP para la LSM.....	23
2.4.1 Realidad Virtual .....	23
2.4.2 Redes Neuronales Artificiales.....	24
2.5 Cálculo Estadístico de Fiabilidad .....	27
<b>CAPÍTULO 3. METODOLOGÍA DE DESARROLLO .....</b>	<b>28</b>
3.1 Metodología.....	28
3.2 Plataforma de Desarrollo .....	29
3.3 Descripción General del Sistema.....	31
3.3.1 Adquisición.....	31
3.3.2 Preprocesamiento .....	34

3.3.3	<i>Segmentación</i> .....	39
3.3.4	<i>Descripción</i> .....	44
3.3.5	<i>Clasificación</i> .....	50
3.4	Interfaz Gráfica .....	58
3.4.1	<i>Implementación en Computadora</i> .....	58
3.4.2	<i>Implementación en Dispositivo Móvil</i> .....	60
<b>CAPÍTULO 4. PRESENTACIÓN Y DISCUSIÓN DE RESULTADOS .....</b>		<b>64</b>
4.1	Resultados del Reconocimiento del Alfabeto Estático de la LSM.....	64
4.2	Análisis de Precisión.....	69
4.3	Limitaciones Encontradas.....	70
4.3.1	<i>Fondo e Iluminación</i> .....	70
4.3.2	<i>Dispositivo para la Captura de Imágenes</i> .....	72
4.3.3	<i>Uso del Descriptor Elegido</i> .....	73
4.4	Comparación de los Resultados .....	75
4.5	Cálculo de Fiabilidad .....	77
4.6	Prueba de Hipótesis .....	79
<b>CONCLUSIONES .....</b>		<b>82</b>
	Aporte computacional.....	83
	Aporte social .....	84
<b>SUGERENCIAS .....</b>		<b>85</b>
<b>REFERENCIAS .....</b>		<b>86</b>
<b>APÉNDICE A.....</b>		<b>92</b>
<b>APÉNDICE B.....</b>		<b>98</b>

## RESUMEN

Hoy día, ha sido posible instaurar una lengua de señas específico para las personas con problemas auditivos como la sordera. Esta lengua de señas se compone de gesticulaciones realizadas con las manos, el cuerpo y el rostro, y que en algunas ocasiones van acompañadas de sonidos. En nuestro país, es denominada Lengua de Señas Mexicana (LSM) y a pesar de la relevancia que tiene respecto al porcentaje de personas con este tipo de discapacidad, hasta la fecha se desconoce la cantidad de población signante, indicando que gran parte de la comunidad mexicana desconoce sobre su existencia, lo que supone la baja o nula capacidad de comunicación con personas sordas. De aquí, que el presente trabajo de investigación hace referencia al empleo del reconocimiento de patrones para el reconocimiento del alfabeto estático de la LSM basado en el uso de los momentos invariantes de Hu considerando 42 imágenes para la etapa de aprendizaje y 2100 para la etapa de pruebas. Para la clasificación de las señas se empleó el clasificador de k vecinos más cercanos. Se realizaron 100 pruebas en ambientes con abundante y poca luminosidad. El porcentaje de precisión en la identificación de las señas es del 72%.

*Palabras clave:* LSM, reconocimiento de patrones, procesamiento de imágenes, momentos de Hu.

## INTRODUCCIÓN

La correcta convivencia y desarrollo social de una persona implica la ejecución de buenas estrategias y herramientas de comunicación. En nuestro país alrededor de 20.8 millones de personas tienen que adecuarse a entornos que en la mayoría de los casos no cuenta con los instrumentos requeridos para su integración al presentar algún tipo de discapacidad. Cerca del 25% de esta porción de la población presenta problemas auditivos (INEGI, 2020).

Como resultado de la necesidad de diálogo con su entorno, se determinó la creación de un idioma propio para las personas de la comunidad sorda que pudiese representar un símbolo de incorporación con la sociedad, es así como surge la lengua de señas. Independiente del espacio geográfico, la lengua de señas se compone de distintas configuraciones realizadas con la mano dominante, acompañadas de gestos faciales y en su caso, sonidos que complementen lo que se requiere transmitir.

Dentro de este marco, a pesar de la relevancia que implica poder comprender esta lengua, no muchas personas tienen conocimiento o interés sobre ella, implicando limitantes de relacionamiento para los individuos en condición de discapacidad auditiva. Visto de esta forma, se busca impulsar la comunicación entre las personas de la comunidad sorda y aquellas que desconocen la Lengua de Señas Mexicana (LSM) por lo que se dispuso la creación de un instrumento tecnológico que pudiese conformar una base para facilitar dicho diálogo.

Por medio del uso de técnicas de Procesamiento Digital de Imágenes, se construyó un sistema que permite el reconocimiento de elementos pertenecientes al abecedario considerado dentro del diccionario de la Lengua de Señas Mexicana, para señas estáticas, con precisión de identificación de 72% para el total de las 21 letras consideradas en ambientes con luz natural y en presencia limitada de esta.

El objetivo de este trabajo de investigación es aportar una herramienta tecnológica que sirva como fundamento y denote la importancia de la inclusión, interacción y comunicación entre las personas sordas y sociedad en general.

Esperando poder representar un apoyo para la comunidad sorda, las personas interesadas en la comunicación en todas sus formas y aquellos quienes demuestren interés en integrar a todas las personas sin importar su condición, se expone el proceso de desarrollo del sistema de reconocimiento de patrones del alfabeto estático de la LSM

con aplicación de los momentos invariantes de Hu y clasificación por medio de los  $k$  vecinos más cercanos.

El presente trabajo se organiza de la siguiente manera: el Capítulo 1 presenta el contexto de la LSM, la problemática abordada, los propósitos específicos de este estudio, el objetivo general, la justificación de este, así como la hipótesis planteada y viabilidad de su abordaje. En el Capítulo 2 se presenta el marco teórico abordando conceptos, teorías y modelos sobre LSM y Reconocimiento de patrones. De manera particular se detalla la metodología planteada por González y Woods (2018), que sirve de base para esta investigación. El Capítulo 3 presenta el desarrollo de la metodología del estudio y su implementación en equipos electrónicos. En el Capítulo 4 se presentan los resultados descriptivos y su diagnóstico. Finalmente, se realiza una discusión de los resultados y se concluye.

## CAPÍTULO 1. PLANTEAMIENTO DEL PROBLEMA

La información conveniente al asentamiento del problema se encuentra en este capítulo, en lo referente al por qué se elige este tema y las posibles herramientas de resolución.

### 1.1 Antecedentes

A lo largo de la historia siempre se ha tenido presente a aquellos grupos de personas que, aunque pueden considerarse una minoría, infieren también sobre el desarrollo de un país, estado y su cultura como sociedad.

Desde la prehistoria, el objetivo más importante para el ser humano ha sido la supervivencia, teniendo como efecto y enfocándose en las personas con algún tipo de discapacidad, el abandono de éstas debido a su situación fisiológica, y por considerarse una carga al emigrar tras la búsqueda de alimento o mejores condiciones de vida.

A pesar de este panorama tan negativo, como refiere Valencia (2014), existen pruebas que a las personas con algún tipo de inconveniente físico o discapacidad se aplicaban distintos métodos de medidas curativas. Así mismo han sido halladas piezas de arte donde es posible apreciar personas con amputaciones o afecciones físicas.

Así mismo, una de las teorías con mayor reconocimiento sobre la influencia de las personas con discapacidad en la historia a nivel mundial es la denominada “tesis de la población excedente que toma en consideración que *“en las sociedades donde la supervivencia económica es precaria, cualquier persona considerada débil o deficiente será eliminada”* (Valencia, 2014 p. 4 - 5).

Sin embargo, dicha teoría ha sido fuertemente impugnada debido a que en las distintas épocas y a pesar de presentarse condiciones económicas cambiantes las personas con discapacidad siguen siendo miembros prevaletentes en la sociedad a nivel mundial.

Por otra parte, y actualmente muy involucrado en las distintas herramientas para el apoyo a las personas con capacidades diferentes, se tiene al *Procesamiento Digital de Imágenes (PDI)* cuyos inicios se mostraron al mundo a manera de perfeccionamiento en las imágenes digitalizadas para un periódico, las cuales serían transmitidas a través de un cable submarino entre Londres y Nueva York. Esta técnica de difusión de imágenes representaba la reducción del tiempo requerido para transportar una imagen a través del océano de alrededor de una semana a aproximadamente tres horas, en la década de 1920 a 1930 (Domínguez, 1996).

Por otra parte, la incorporación de las computadoras en el procesamiento de imágenes tuvo comienzo en el Jet Propulsion Laboratory en 1964 (O'Handley y Green, 1972) al ser transmitidas las imágenes de la Luna por el Ranger 7, encargándose la computadora de corregir las distorsiones variantes en los sistemas de captura de la cámara de televisión. Estos procedimientos en conjunto dieron pie a las bases para la mejora de los anteriores métodos utilizados para la restauración y mejora de las imágenes.

Posteriormente, alrededor de los años 70, y con el desarrollo de los microprocesadores, pudo amplificarse el alcance del procesamiento y, por tanto, el almacenamiento de las imágenes. En estas mismas épocas se creó un sensor con células fotoeléctricas, que registraba una imagen, permitiendo así el nacimiento de las cámaras digitales.

De esta forma, en 1975 se presentó la primera cámara digital presentada por Kodak y, posteriormente, solo un año después se pudo captar imágenes a color (Gómez, 2017). Es así como las diferentes líneas de generación de conocimiento que se han desarrollado para el reconocimiento de lengua de señas pueden delimitarse a la forma a como son obtenidas las características de entrada, siendo estas:

#### *1. Métodos basados en obtener impulsos generados por guantes.*

Hernández (2019), utilizó diferentes sensores colocados en un guante inteligente para generar impulsos a partir de la orientación, posición y ángulos de la mano. Estos impulsos son trasladados a mediciones que en conjunto determinan el tipo de señal realizada.

#### *2. Métodos basados en visión.*

Estos métodos son desarrollados a partir de obtención de imágenes en cámaras, que pasan por diversos procesamientos hasta obtener las características que definen a cada seña mediante forma, longitud de la mano, posición, flexión y ángulos tanto de la mano como de los dedos. Se consideran como métodos "*no invasivos*", puesto que no es necesario el uso de algún tipo de hardware en específico (Trujillo y García, 2021).

Con respecto a la aplicación de dichos métodos, Barragan et al. (2013) resaltaron la importancia de las características únicas que tiene la LSM, y sobre todo el hecho de que estas sean inherentes al lenguaje propio de México hacen que sea difícil extrapolar una solución ya existente al mismo, por lo que es importante contar con una solución que contemple la estructura gramatical única con la que cuenta.

Así mismo, Hernández de la Luz et al. (2018), llevaron a cabo un análisis de tecnologías y una arquitectura de un prototipo intérprete de LSM, apoyado en dispositivos de captura



de movimiento 3D, con el objetivo de que sirva de base para el desarrollo de un intérprete que brinde una mejora sustancial en la calidad de vida de las personas con discapacidad auditiva, al permitir comunicarse con el resto de la sociedad. De aquí la importancia de desarrollar una arquitectura computacional de interacción para la gente sorda, destacando el hecho de que, en México, a pesar de existir legislaciones que buscan promover la integración de las minorías en la sociedad, se necesitan herramientas y propuestas que ayuden a mejorar la inclusión.

Por su parte, Borja Araiza et al. (2019), desarrollaron una herramienta visual con la finalidad de permitir el aprendizaje de múltiples palabras, haciendo uso del lenguaje Virtual Reality Modeling Language (VRML) y de una base de datos a partir de movimientos y posiciones establecidos por la LSM. Dentro del estudio se hizo un especial énfasis sobre la variación léxica existente en esta, puesto que no tienen una clase léxica en específico, tienen diversos comportamientos prototípicos que funcionan como sustantivos, verbos, nominales o modificadores, adjetivos o adverbios, y a su vez una infinidad de conceptos dependiendo del contexto de uso, por lo que es importante identificar no solo el contexto, sino el uso de las expresiones, acciones, espacios, entidades y la categoría gramatical que referencian a la persona de la que se habla.

Dado lo anterior la LSM es tan variado como cualquier lenguaje, convirtiéndose este en un campo de estudio bastante amplio y poco explorado por lo que surge la imperante necesidad de estudiar las diversas manifestaciones de estas variaciones lingüísticas no solo enfocándose al país en general, sino profundizando en cada región por las distintas variaciones adoptadas por los señantes.

De igual forma, Hernández de la Luz (2019), implementó un sistema intérprete de Lengua de Señas Mexicana con vocabulario configurable según el contexto, cuyo propósito fue principalmente el reconocimiento de las señas pertenecientes a la LSM a través de dispositivos de captura de movimientos en tercera dimensión. Este intérprete consiste en el procesamiento de las imágenes obtenidas por los dispositivos señalados de detección de movimientos en 3D que conllevan tiempos de 500 a 900 milisegundos.

Por su parte, Trujillo y García-Bautista (2021), efectuaron el reconocimiento de lengua de señas a través del análisis de la mano dominante, los dedos y expresiones faciales mediante un sistema basado en sensores RGB que permiten obtener la información fotométrica perteneciente al esquema de la imagen. Con el empleo de un Corpus de la LSM (CLSM), que es una base de datos con la información requerida para el desarrollo de sistemas de reconocimiento de señas, se obtuvieron los datos correspondientes a

imágenes en tercera dimensión como la profundidad, el color, las nubes de puntos, así como 14 articulaciones del esqueleto humano.

Es así, que, con base en los trabajos mencionados, la realización de este proyecto implicó el desarrollo de un sistema que provee de una herramienta base para el reconocimiento de patrones del alfabeto estático de la LSM, mediante la aplicación de los momentos invariantes de Hu y con el empleo de dispositivos de fácil acceso, tales como lo son una computadora o un teléfono celular.

## **1.2 Definición del Problema**

Dentro de la sociedad en donde convivimos existen sectores considerados vulnerables, siendo estos la población que presenta algún tipo de discapacidad, enfocándonos en personas sordas, que presentan poca o nula integración y que intentan ejercer sus derechos para tener un desarrollo social en balance. Estos sectores en comparación al total de la población conforman un porcentaje significativo.

Por ejemplo, en el Censo del 2020, el INEGI contó en el rubro de discapacidad a 20 millones 838 mil 108 personas, una cifra que representa el 16.5% de la población de México. Esta cifra resulta de la suma de los 6 millones 179 mil 890 (4.9%) que fueron identificadas como personas con discapacidad, más los 13 millones 934 mil 448 (11.1%) que dijeron tener alguna limitación para realizar actividades de la vida diaria (caminar, ver, oír, autocuidado, hablar o comunicarse, recordar o concentrarse), y a los 723,770 (0.6%) con algún “problema o condición mental”, éste último, un término que permite equiparar la medición con la del Censo 2010, en la que se hablaba de personas con “limitación mental” (Dis-capacidad, 2021).

Adicionalmente, sobre el tipo de limitaciones reportadas en el Censo 2020, llama la atención la cantidad de personas que no pueden ver, aun usando lentes, que es ligeramente superior a la cantidad de personas con limitación para caminar, o discapacidad motriz. Es así como, algunos de los problemas de discapacidad pueden ser parcial o totalmente resueltos mediante algún tipo de prótesis, código Braille o con apoyo de animales entrenados para servir de guía.

Así mismo, para personas con discapacidad auditiva (24.4% del total de la población con alguna discapacidad en México) se desarrollaron lenguas de señas que son utilizadas y enseñadas desde la educación básica, como lo es el Diccionario de Lenguas Mexicano elaborado por el Consejo Nacional para Prevenir la Discriminación (Conapred, 2020).

Este diccionario contempla que las señas se realizan con la mano dominante, es decir, si la persona es diestra, con la mano derecha y si es zurda, con la izquierda. Las señas son realizadas con la mano derecha y al colocarse el libro sobre el pecho, entonces, las imágenes corresponden a la posición de esa mano, de modo que las personas zurdas deberán hacer algunos ajustes (Serafín de Feischmann y González, 2011)

Por tanto, con el Diccionario de Lenguas Mexicano se facilita el manejo de este sistema de comunicación, se fomenta una cultura incluyente y se reafirma la diversidad del patrimonio cultural de nuestro país. Es así, que las propuestas de proyectos de reconocimiento de lengua de señas hasta ahora desarrolladas se centran en el mero procesamiento de imágenes o en el uso de sensores para el reconocimiento del movimiento de las manos, dando como resultado herramientas potentes pero que dependen del procesamiento en el momento en que se lee la seña predispuesta.

Sin embargo, el reconocimiento de la lengua de señas enfrenta problemas característicos de los sistemas de reconocimiento automatizados de objetos en cualquiera de sus etapas: adquisición de imagen, preprocesamiento de la imagen, segmentación de las manos, detección y extracción de características relevantes de las manos, descripción e interpretación de las señas expresadas con las manos.

Por ejemplo, en la etapa de adquisición se presentan problemas como elegir un dispositivo adecuado a la aplicación considerando la resolución de la imagen adquirida y velocidad de su muestreo, esto debido a que existen diferentes tipos de equipos con la capacidad de adquisición. Referente al problema de preprocesamiento se refiere a la propia configuración del equipo o dispositivo que varía de acuerdo con el fabricante. Para la fase de segmentación se presentan problemas debido a la iluminación, enfoque, distancias, encuadre y velocidad de captura. De igual manera, se presentan problemas por variación de la luminosidad que afecta en los colores y esto dificulta el proceso de segmentación. También influyen condiciones como el clima y posición del objeto creando distorsiones que dificultan la extracción de características relevantes para el sistema.

De aquí, que esta problemática se vincula a áreas tales como tratamiento de imágenes, reconocimiento de patrones, visión artificial y minería de datos, así como estadística, por tanto, se justifica la creación de una herramienta digital que permita la traducción de la LSM para colaborar en la comunicación entre las personas sordas y las personas oyentes, a través de una plataforma móvil, es decir, el uso de un dispositivo celular o computadora.

Es entonces que se resaltan los problemas o dificultades que presentan las personas en condición de discapacidad auditiva, por lo que es de vital importancia profundizar en ofrecer un mecanismo de comunicación efectivo que les permita desarrollarse inclusivamente y comunicarse con cualquier individuo a pesar de que este no tenga ningún conocimiento sobre la LSM.

Por otra parte, con respecto a los equipos móviles con sistema operativo Android, se determinó su empleo para esta investigación ya que hasta el año 2020 lideraba el número de usuarios en México, con un 84.6% en contraste con el 6.8% de la población mexicana que utilizaba iOS. Resaltando que las personas encuestadas consideraron no realizar cambio del sistema Android al perteneciente a Apple (IFT, 2022).

### **1.3 Importancia del Tema**

Tras el estudio y análisis acerca del trasfondo social sobre la importancia del acercamiento e inclusión de las personas con discapacidad auditiva, se ha definido el objeto de esta investigación el uso e implementación de la LSM de manera digital y con apoyo de la tecnología con el propósito de fungir como herramienta de comunicación entre el público general y las personas que presenten este tipo de “obstáculos” para su desarrollo de manera íntegra.

El objeto de llevar a cabo esta investigación es conocer el estado y condiciones sobre la actual comunicación para las personas con discapacidad auditiva y cuáles han sido las aportaciones para crear una sociedad equitativa con estas personas. Por medio de la recopilación de esta información, se busca también aportar una herramienta tecnológica enfocada en fomentar la comunicación entre personas signantes y aquellas que no practican la LSM.

### **1.4 Pregunta de Investigación**

Una pregunta es el inicio y el eje de la investigación, no es solo un asunto semántico o de redacción. La pregunta parte del problema de investigación que es una brecha en el conocimiento entre lo que es en la realidad y lo que debería ser, es algo que debemos resolver como investigadores. Es así, que, para apoyar en la resolución de esta problemática, esta investigación pretende dar respuesta a la siguiente interrogante:

*¿La aplicación de los momentos invariantes de Hu permiten el reconocimiento de patrones del alfabeto estático de la LSM con implementación en dispositivos móviles?*

## **1.5 Alcances del Problema**

### **1.5.1 Contextual**

Las señas consideradas para este trabajo se refieren a las obtenidas del “Diccionario de Lengua de Señas Mexicana” haciendo referencia al primer apartado de dicho documento, siendo estas las señas oficiales consideradas *estáticas del alfabeto de la LSM* (no presentan movimiento requerido para su comprensión), tomando en cuenta el establecimiento de diálogo entre personas sordas y aquellas que no presentan esta discapacidad.

Así mismo y como se ha mencionado, únicamente la LSM es empleada para el desarrollo del presente trabajo.

### **1.5.2 Espacial**

La presente investigación se realizó en Toluca Estado de México, realizando 100 pruebas en ambientes con y poca presencia de luz en la Facultad de Ingeniería de la Universidad Autónoma del Estado de México. Las señas fueron ejecutadas por personas con conocimiento de la LSM.

### **1.5.3 Temporalidad**

El presente trabajo tuvo una duración de octubre 2021 a octubre de 2023, debido a que durante el desarrollo de esta investigación se combinaron actividades laborales.

### **1.5.4 Semántica o Disciplina**

Este trabajo de investigación se centra en la ciencia de Inteligencia Artificial, específicamente en las disciplinas de reconocimiento de patrones y procesamiento digital de imágenes; así mismo el área de Reconocimiento de Lengua de Señas Mexicana (Escritura de Signos).

## **1.6 Objetivos**

### **1.6.1 Objetivo General**

Desarrollar un sistema de reconocimiento de patrones del alfabeto estático de la LSM, mediante la aplicación de los momentos invariantes de Hu con el fin de fomentar la

inclusión, interacción y comunicación con personas que presentan discapacidad auditiva.

### **1.6.2 Objetivos Particulares**

- a) Reconocer y capturar imágenes estáticas por medio de la cámara del dispositivo móvil con presencia de luz.
- b) Mejorar la nitidez y los elementos de la imagen, así como eliminar aquellos rasgos indeseables para un mejor procesamiento de esta.
- c) Identificar la seña captada en la imagen con base en los patrones almacenados en la Base de Conocimientos (BC).

### **1.7 Justificación**

Tras el desarrollo e implementación de esta investigación y el sistema en cuestión, se plantea el beneficio sobre la incentivación en la comunicación de las personas que presentan discapacidad auditiva, permitiendo mejorar el desarrollo, desenvolvimiento y crecimiento personal de las personas con estas características.

Con la elaboración de la presente investigación se supone, además, la añadidura de partes significativas a los anteriores intentos e implementaciones de sistemas, programas y desarrollos enfocados en la inclusión de las personas con problemas auditivos, formando parte de la amplia gama de investigaciones que buscan proveer de información valiosa a las futuras creaciones relacionadas con el rubro auditivo. A través de la forma de implementación que se propone en este documento, mediante el lenguaje de programación elegido, el método de gestión de la información y el tratamiento de las imágenes, la extracción de las debidas características de estas y la presentación de los resultados obtenidos, se idea también un material de actualización respecto a los trabajos realizados con anterioridad de acuerdo con las tecnologías disponibles hasta el momento de la realización de éste.

Así mismo se diseñó una BC para el reconocimiento de patrones del alfabeto de la LSM con el propósito de interpretar esta lengua de manera más completa en el futuro.

En principio, se propone el uso de la herramienta en espacios de convivencia entre personas con incapacidad auditiva comenzando por el hogar, talleres de LSM, fundaciones o asociaciones que apoyen a personas con este tipo de características, etc.

Adicionalmente, acrecentar el uso del sistema ampliando las características conforme a la usabilidad y porcentaje de ajuste a los diccionarios, palabras y frases usadas por las

personas signantes. Como el sistema representa una herramienta de comunicación básica entre signantes y no signantes centrándose en la dactilología de la LSM, en un período no mayor a cinco años se conjetura el uso de este en espacios de mayor alcance.

### **1.7.1 Institucional**

El presente trabajo marca un precedente de empatía, conocimiento, diversidad lingüística y desarrollo tecnológico dentro de la Facultad de Ingeniería de la Universidad Autónoma del Estado de México. Motivando a los estudiantes, docentes y público en general, a introducirse en la creación de tecnologías que apoyen el desarrollo integral de sectores vulnerables de la sociedad mexicana.

Aunado a lo anterior, se reconoce la meritoria aportación al ser una de las Universidades de México que pone su atención en la problemática de comunicación y desarrollo de las personas que presentan cualquier tipo de discapacidad en este caso la auditiva, siendo una contribución a la inclusión en centros educativos como se está realizando en el Instituto Tecnológico de Toluca (ITT) (Sol de Toluca, 2023).

### **1.7.2 Profesional**

La aportación profesional que esta investigación provee en el campo de la Ingeniería en Computación se enfoca en el uso de las herramientas, códigos y algoritmos que el Ingeniero en Computación ha formado, desarrollado, previsto e implementado a lo largo de su formación académica.

Proporcionando de esta forma un aporte tecnológico a partir de los conocimientos generados a lo largo de su trayectoria académica, además de contribuir con la elaboración de un sistema que permita mejorar la convivencia de las personas que presentan discapacidad auditiva, con la sociedad.

### **1.7.3 Personal**

El presente trabajo de investigación tiene una aportación relevante dentro del campo humanístico, gracias al desarrollo del sistema para el reconocimiento del alfabeto estático de la LSM puesto que permite que cualquier persona tenga acceso a una comunicación sencilla, rápida y práctica con personas en situación de discapacidad auditiva a través del empleo de la dactilología, aun cuando estas no tengan algún conocimiento sobre el diccionario de la LSM.

Influyendo así de manera favorable, ya que la implementación de dicha aplicación promueve la inclusión de las personas con este tipo de discapacidad, con el objetivo de ofrecer mejores servicios a esta parte vulnerable de la población, permitiéndoles ejercer plenamente sus derechos y apoyando la convivencia en balance de la sociedad.

Además de la valiosa aportación de inclusión social, mantiene la preservación del patrimonio lingüístico, pero sobre todo fomenta la empatía con aquellos que presentan hipoacusia leve o aguda dando como resultado una participación completa sin ningún tipo de barrera o distinción.

Se presenta entonces este trabajo y el sistema obtenido como una oportunidad de ampliar el panorama de oportunidades para las personas con discapacidad auditiva, así como el del público en general con el fin de crear ambientes y espacios incluyentes sin preferencias o características parcialmente favorables para los distintos sectores de la sociedad mexicana.

## **1.8 Hipótesis**

*El reconocimiento de patrones del alfabeto estático de la LSM esta positivamente asociado con el uso de momentos invariantes de HU.*

La hipótesis aquí planteada es de *tipo descriptiva y correlacional*, cuya comprobación apoyará en la resolución de la pregunta de investigación de este trabajo mediante el cálculo de la correlación de *Pearson* a través del software *IBM SPSS*.

## **1.9 Viabilidad**

Para el desarrollo de la presente investigación se empleó el lenguaje de programación *Python* y el Framework *Kivy*,. Ambas son herramientas de uso gratuito y fueron empleadas con el propósito de realizar las pruebas pertinentes y su implementación en equipos con el sistema operativo *Android*.



## CAPÍTULO 2. MARCO TEÓRICO Y ESTADO DEL ARTE

Este capítulo aborda la información relacionada a los términos utilizados en este trabajo, infiriendo sobre las definiciones de Discapacidad Auditiva, Lengua de Señas Mexicana, Reconocimiento de Patrones y como estos últimos, han permitido crear herramientas para la integración de personas de la comunidad sorda.

### 2.1 Discapacidad Auditiva

El Sistema Nacional DIF define a la discapacidad auditiva como “la falta, disminución o pérdida de la capacidad para oír en algún lugar del aparato auditivo y no se aprecia porque carece de características físicas que la evidencien” (Sistema Nacional DIF, 2017)

La discapacidad auditiva es una limitante que requiere ser atendida por la sociedad, para que a las personas con esta afectación se les permita garantizar integración, éxito y autorrealización. De aquí surge la necesidad de desarrollar herramientas y abrir lugares que promuevan la inclusión, empezando por el sector educacional ya que se requiere de prácticas y actividades de compañerismo inclusivo, así como de herramientas comunicativas que generen un valor educacional activo.

Investigaciones recientes han remarcado la importancia de atender esta discapacidad. En primer término, se hace referencia a la investigación *Estrategia integral para la inclusión con las personas en condición de discapacidad auditiva, una apuesta en el marco de la educación* realizada por Quevedo et. al en 2019 en Colombia, cuyo propósito es exponer la importancia de la unión de los contextos donde se desenvuelven personas con discapacidad auditiva, dado que esta unión propicia espacio de mayor desarrollo y que también promueve su participación dentro de comunidades, y ello les permite ser parte activa dentro de ellas.

Este estudio se realizó bajo un enfoque cualitativo. La metodología utilizó una secuencia de etapas estratégicas para generar la inclusión social de las personas en condición de discapacidad auditiva. La primera etapa es la encargada de identificar las principales características de la familia, institución o comunidad que favorecen o limitan el proceso, con la finalidad de establecer acciones que promuevan la equidad, justicia y respeto a la diversidad. En la segunda etapa se busca planificar aquellas acciones que promuevan y garanticen un acompañamiento integral en todas las personas en especial en miembros sordos. En la tercera etapa se debe aplicar una estrategia de intervención que permita la participación de los involucrados, así como la disposición al cambio y

retroalimentación constante. Por último, la cuarta etapa se enfoca en evaluar y analizar los resultados tomando en cuenta los criterios de cada etapa y midiendo el grado de satisfacción de los involucrados de forma válida, confiable y pertinente.

Ahora bien, según el reporte de matrícula del Ministerio de Educación Nacional de Colombia correspondiente al 2008:

“Se encuentran 8.977 estudiantes con sordera profunda e hipoacusia: la mayoría se encuentra en básica primaria (49%): de este grupo de estudiantes, presentan sordera profunda (63%) y un 96% de estudiantes identificados en el grupo de víctimas del conflicto (318 estudiantes), están en situación de desplazamiento” (INSOR, 2009).

Dichas cifras llevan a reflexionar si los docentes y la sociedad están preparados para afrontar, incluir y preparar a personas en condición de discapacidad auditiva, por tanto, este estudio sugiere un compromiso institucional mediante la implementación de las cuatro etapas previamente descritas (caracterización del contexto, planificación de un conjunto de acciones, aplicar estrategias de intervención y evaluación del procesos y análisis de resultados).

Los autores concluyen que es responsabilidad de la sociedad desarrollar estrategias para fomentar un aprendizaje innovador, promover la participación de todos los miembros de la sociedad, hacer uso de las TIC para desarrollar herramientas que mejoren la comunicación y con ello evitar el aislamiento y bajo desarrollo de integración de aquellos con discapacidad auditiva y que radican en lugares alejados.

La discapacidad auditiva, como otras discapacidades han representado un obstáculo para el correcto desenvolvimiento de las personas en la sociedad, tal como lo deducen Santa Cruz et al. (2021), quienes destacan que el presentar algún tipo de discapacidad indica posibles deficiencias para desarrollarse en el entorno escolar.

La investigación antes mencionada, fue realizada bajo un enfoque cuantitativo con un diseño de estudio no experimental transversal de tipo correlacional que plantea un medio de comparación para poder evaluar el desarrollo sobre competencias, adaptación y los posibles problemas que pueden presentar los niños en etapa preescolar con discapacidad parcial o total, ya sea visual, auditiva o sensorial en relación de los que manifiestan un desarrollo típico. Aplicado a 192 estudiantes, siendo un 47.91% del sexo masculino con una edad promedio de 11.178 meses.

Los resultados obtenidos delegan que los alumnos que presentan discapacidad auditiva superan estadísticamente el índice de problemáticas en comparación con aquellos que presentan discapacidad auditiva o los de desarrollo típico, infiriendo sobre el

aprendizaje, estados de ánimo, sociabilidad, problemas de pensamiento, problemas de atención, entre otros factores. Así mismo, se evidencia que estos resultados coinciden con los expresados por Van Eldik et al. (2004).

En sus conclusiones se deduce cómo pueden llegar a verse afectados los infantes al iniciar el historial escolar cuando presentan discapacidad sensorial, visual o auditiva con el fin de proporcionar información estadística cuyo objetivo pueda permitir la implementación de medidas de prevención para los alumnos con estas características, y así mismo reducir los problemas emocionales que derivan en problemáticas aún mayores en el ámbito académico y social.

## **2.2 Lengua de Señas Mexicana (LSM)**

La literatura define Lengua de Señas Mexicana (LSM) como recursos comunicativos que los señantes organizan entorno a las condiciones socioculturales. El Diario Oficial de la Federación (DOF, 2011) la define como “La lengua de una comunidad de sordos, que consiste en una serie de signos gestuales articulados con las manos y acompañados de expresiones faciales, mirada intencional y movimiento corporal, dotados de función lingüística, forma parte del patrimonio lingüístico de dicha comunidad y es tan rica y compleja en gramática y vocabulario como cualquier lengua oral”.

La LSM no es considerado como una etnia a pesar del gran número de usuarios, por no cumplir con ciertos criterios, tales como rituales, costumbres y un territorio propio. Sin embargo, su inclusión de la LSM como lengua Nacional en 2003 ha promovido su enseñanza, logrando incrementar cada vez más el número de hablantes.

La investigación *La Lengua de Señas Mexicana, sus hablantes y su estructura* realizada por Escobar en 2017, busca exponer la importancia de la LSM por no poseer el mismo canal comunicativo con alguna otra lengua dominante como el español, además de tener diversas variaciones léxicas dependiendo de su ubicación geográfica y sociedad en general, generando consigo una gran diversidad de señas secundarias adoptadas por los señantes.

Dicho estudio se realizó bajo un enfoque cualitativo. La metodología utilizó tres condiciones de variación, la primera identifica aquellas señas que se refieren a conceptos similares, pero con distinta estructura articulatoria, la segunda hace alusión a las señas con articulación idéntica usadas para expresar conceptos distintos, y en tercer lugar reconoce conceptos que son nombrados con señas que no existen en otras variantes.

Al existir una diversidad léxica entre los hablantes de la LSM, es de vital importancia reconocer la variación morfológica, la gramática, el orden sintáctico y semántico, así como las señas temporales, o únicas de cada región. Para ello se tomaron en consideración dos factores importantes propuestos por Robins (1964), siendo el primer factor la *clase cerrada* y el segundo la *clase abierta*. Robins identifica como clase cerrada:

“... contienen un número fijo y, usualmente, pequeño de palabras que son las mismas para todos los hablantes de una lengua o dialecto y que no pierden o agregan miembros sin alterar estructuralmente a la gramática de la lengua en su totalidad.” (p. 230)

Partiendo de lo anterior, la LSM no tiene una clase léxica en específico, tienen diversos comportamientos prototípicos que funcionan como sustantivos, verbos, nominales o modificadores, adjetivos o adverbios, y a su vez una infinidad de conceptos dependiendo del contexto de uso. Además de la diversidad léxica de la LSM, esta cuenta *con usos no verbales sin cambios en la estructura articulatoria*, es decir carecen de predicado en las oraciones, por lo que es importante identificar no solo el contexto, sino el uso de las expresiones, acciones, espacios, entidades y la categoría gramatical que referencian a la persona de la que se habla.

En general, los autores señalados concluyen que las condiciones de aprendizaje de la LSM varían entorno a diversas situaciones y distintos factores que siguen siendo un campo poco explorado, del cual se requiere tomar conciencia de las distintas formas de manifestación de estas variaciones lingüísticas, mismas que se pueden representar dependiendo de las estrategias de comunicación tanto de forma articular como gestual, formando parte de la pragmática, misma que puede ser representativa de una o varias regiones en específico, o simplemente una seña temporal deíctica, y pronominal.

La lengua de señas ha representado un apoyo a gran escala en cuanto a comunicación para las personas con discapacidad auditiva, considerándose más transparente en lo que a iconicidad se refiere en comparación con las lenguas orales como lo expresa Moreno (2018). Sin embargo, en ocasiones no suele manifestarse de esta forma clara.

Definiendo a la iconicidad, en términos lingüísticos, como la relación de motivación entre el significante y el significado.

Con la aplicación de un paradigma cualitativo, se precisa el impacto de la lengua de señas de manera general, a partir de enfoques aplicados en el estudio de las lenguas naturales siendo estos: enfoque filológico que se centra en la lengua escrita, enfoque

oralista que refiere a la lengua oral como objeto legítimo y el enfoque naturalista cuyo objetivo fue incluir a las lenguas de señas como objeto legítimo independiente.

Con el estudio de las lenguas de señas, se pudo determinar que no solamente las lenguas escritas y orales cumplen con los enfoques previstos, y que además al hacer uso de lenguas signadas se configuran operaciones mentales que tienen su desarrollo desde la percepción del objeto o seña, hasta su comprensión considerando así también a esta investigación como una aportación en la caracterización de las estructuras y procesos mentales considerados en la competencia lingüística.

Finalmente, se concluye que tanto lenguas orales como signadas presentan características iguales, desde su composición hasta su presentación. Conllevando también a la aportación mediante el estudio de las lenguas de señas de importantes conceptos en el conocimiento de la facultad humana de lenguaje. Por lo que, el impacto considerado dentro del desarrollo de la lingüística se verá fortalecido para tiempos venideros.

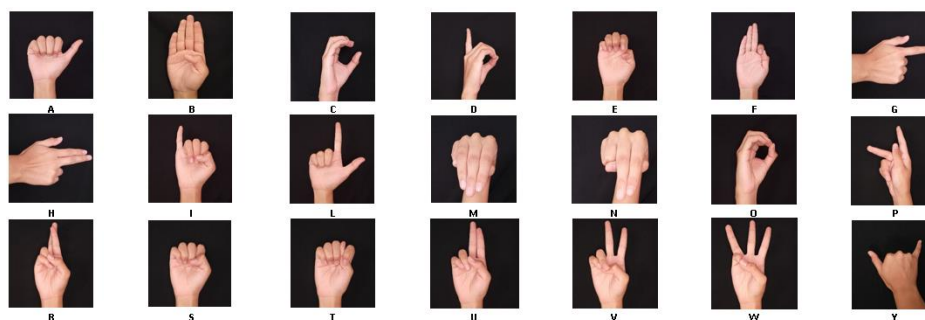
Parte importante del aprendizaje de la LSM y considerado elemento principal para su comprensión es el alfabeto también determinado como dactilología, ya que como mencionan Bustos-Rubilar, et al. (2021), “(...) funciona como un puente entre la lengua de señas y la lengua oral, ya que ambas comparten información gesto-visual”.

De otra manera, como indica Cruz y Serrano en 2014, la dactilología es un instrumento útil para formar palabras con la lengua de señas, basándose en la representación manual de las letras del alfabeto correspondiente, en este caso a la LSM.

Por otra parte, las señas del alfabeto consideradas como estáticas son aquellas que no presentan algún tipo de movimiento para su identificación, visibles en la figura 2.1, centrándose en 21 de las 27 letras correspondientes al abecedario mexicano.

**Figura 2.1**

*Alfabeto estático de la LSM*



### 2.3 Reconocimiento de Patrones (RP)

Se denomina Reconocimiento de Patrones (RP) a la identificación y clasificación de objetos o comportamientos a partir de la obtención de sus características principales o elementos recurrentes. Su uso tiene mayor incidencia en las ramas de matemáticas e informática, con enfoque en aplicaciones de visión artificial y detección de objetos (MathWorks, s.f.).

El RP se compone de los elementos listados a continuación.

**Reconocimiento:** Distinción o identificación de un objeto entre una agrupación establecida, este objeto posee características y atributos propios que lo diferencia del resto (CEUPE, s.f.)

**Patrón:** Elemento que presenta cierto comportamiento o características ya conocidas.

**Clase:** Conjunto de objetos cuyas características o comportamientos son similares.

**Atributo:** Característica, medida o cualidad extraíble.

El proceso de RP usualmente se define por medio de cuatro fases fundamentales: adquisición de datos, preprocesamiento, extracción de características y clasificación. Una vez que se ha obtenido el conjunto de datos, estos se preprocesan para etapas posteriores. Después se aplica la extracción de características, convirtiendo la información perteneciente al conjunto de datos previo, en vectores que representan los datos originales. Esta información es empleada en la etapa de clasificación con el fin de separarla en clases (Karyakarte y Savant, 2019). En la figura 2.2 se presenta el modelo básico de RP.

**Figura 2.2**

*Modelo básico del RP*



*Nota.* Adaptado de “Pattern Recognition Process, Methods and Applications in Artificial Intelligence” (p. 1163), por S. Karyakarte y I. Savant, 2019, *International Research Journal of Engineering and Technology (IRJET)*, 6(11).

- a) **Adquisición de datos:** Se recaban los datos a ser analizados, comúnmente a través de un dispositivo denominado *sensor*. Las variables obtenidas pueden ser

de distintas índoles, por ejemplo: color, temperatura, intensidad lumínica, inclinación, velocidad, fuerza, entre otras (CEUPE, s.f.).

- b) Preprocesamiento:** El patrón de interés es segmentado del fondo, con el objetivo de obtener datos más consistentes y la reducción de variaciones. Se incluyen procesos de corrección de imagen (reducción de ruido, suavización, etc.), así como variaciones de iluminación en cuanto a dirección e intensidad (Karyakarte y Savant, 2019).
- c) Extracción de características:** Se crea el vector de características a partir de la transformación de los datos a una versión reducida. Únicamente se extrae la información de interés, con el propósito de llevar a cabo el análisis para el tema o actividad de enfoque a partir de la representación reducida, en vez del tamaño original de los datos.

En esta etapa se pueden utilizar dos tipos de características:

**Características con connotación física:** geométricas o estructurales y estadísticas. No abordan aquellas que son irrelevantes.

**Características sin connotación física:** denominadas características de mapeo. Debido a la definición de límites entre clases son claros, hacen que la clasificación sea más fácil; sin embargo, la complejidad a nivel computacional es mayor.

(Karyakarte y Savant, 2019)

- d) Clasificación:** Se utilizan las características previamente extraídas para asociarlas de acuerdo con los rasgos comunes, identificándolas y, asimismo, asociándolas con la clase apropiada. Tecnologías de *machine learning* son empleadas, cuyo objetivo es hacer que las computadoras *aprendan* a clasificar los objetos de manera autónoma. Este proceso considera alguno de los siguientes enfoques.

**Supervisada:** Plantea la correspondencia entre la información de entrada, etiquetada de manera manual, en contraste con los resultados deseados. Se supone el uso de datos etiquetados de gran tamaño.

Se emplea un conjunto de datos conocidos, previamente etiquetados que se emparejan con los resultados esperados, con la instauración de relaciones entre las características y los datos de salida para determinar los valores de respuesta.

De manera general, utiliza un conjunto de elementos como ejemplo, a los que ya se les ha asignado la etiqueta de clase predeterminada; es usada

para propósitos de clasificación. Ejemplo es el **algoritmo de k vecinos más cercanos**, también conocido como **KNN o k-NN**. Es un clasificador de aprendizaje supervisado no paramétrico, que utiliza la proximidad para hacer clasificaciones o predicciones sobre la agrupación de un punto de datos individual

**No supervisada:** Propone la búsqueda de estructuras ocultas en datos no etiquetados, para conjuntos que no son de gran tamaño. Las técnicas de RP aplicadas se especializan en aquellas actividades en las que se dificulta la obtención de datos suficientes para la detección de objetos.

De manera general, propone el encontrar características inherentes en la estructura de la información; es usada para propósitos de agrupamiento.

(Narasimha, M. y Susheela, 2015; Karyakarte y Savant, 2019; Sharma y Kaur, 2013; MathWorks, s.f.)

**Clasificación parcialmente supervisada:** Usa una pequeña cantidad de muestras de datos etiquetadas, y un gran número de información sin etiquetar para que el clasificador pueda ser entrenado (Narasimha, M. y Susheela, 2015).

## **2.4 Técnicas y Métodos Basados en el RP para la LSM**

### **2.4.1 Realidad Virtual**

Borja et al. (2019) definen realidad virtual como *“una representación de la realidad a través de medios electrónicos, que nos da la sensación de experimentar una situación real, en la que podemos interactuar con lo que nos rodea”*. El objetivo de estos autores se basó en desarrollar una herramienta visual, que permite el aprendizaje de múltiples palabras que forman parte de la LSM, haciendo uso del lenguaje VRML y de una base de datos a partir de movimientos y posiciones establecidos.

La metodología utilizada fue el *Rapid Application Development* (RAD) por James Martin en 1980, siendo este un método que comprende el desarrollo iterativo, la construcción de prototipos y el uso de utilidades CASE.

Partiendo de lo anterior, la LSM no tiene una clase léxica en específico, tiene diversos comportamientos prototípicos que funcionan como sustantivos, verbos, nominales o modificadores, adjetivos o adverbios, y a su vez una infinidad de conceptos dependiendo



del contexto de uso. Además de la diversidad léxica de la LSM, esta cuenta *con usos no verbales sin cambios en la estructura articulatoria*, es decir carecen de predicado en las oraciones, por lo que es importante identificar no solo el contexto, sino el uso de las expresiones, acciones, espacios, entidades y la categoría gramatical que referencian a la persona de la que se habla.

Los autores concluyen que las condiciones de aprendizaje de la LSM varían entorno a diversas situaciones y distintos factores que siguen siendo un campo poco explorado, del cual se requiere tomar conciencia de las distintas formas de manifestación de estas variaciones lingüísticas, mismas que se pueden representar dependiendo de las estrategias de comunicación tanto de forma articular como gestual, formando parte de la pragmática, misma que puede ser representativa de una o varias regiones en específico, o simplemente una seña temporal deíctica, y pronominal.

#### **2.4.2 Redes Neuronales Artificiales**

Las redes neuronales artificiales (ANN por su traducción al inglés), se definen como un subconjunto de *machine learning* que son base fundamental de los algoritmos de *deep learning*. Estas se constituyen por capas de nodos con una para la entrada, varias capas ocultas y una de salida, cada uno de los nodos (también denominados neuronas artificiales), se conecta a otro, con la asignación de un peso y umbral correspondientes.

Estas redes se emplean para el entrenamiento de datos con el objetivo de aprender y mejorar su precisión con el tiempo, cuando se alcanza el nivel de ajuste precisado, se utilizan como parte de herramientas de reconocimiento y clasificación de datos, a gran velocidad (IBM, s.f.).

En primer lugar, con el empleo de la tecnología *Leap Motion*, Cuecuecha et al. (2015) desarrollaron un sistema de reconocimiento de la LSM con el objetivo de reconocer las vocales signadas que son de tipo estático con el uso de dispositivos con sensores de reconocimiento de gestos, tal como el reconocido *Kinect*.

Con la aplicación del paradigma cuantitativo se realizó el estudio sobre el reconocimiento de las vocales, haciendo énfasis en las posiciones de los dedos, ya que la interpretación de los gestos depende únicamente de la configuración espacial de éstos. De esta forma, se obtuvieron distancias que implican la distinción de las diferentes letras, con la recolección de 100 muestras, es decir, 20 para cada letra signada desempeñando el papel de base de aprendizaje y validación del reconocimiento de las vocales. Para dicho proyecto la clasificación se realizó con el sustento de una red

neuronal artificial, entrenada con perceptrón multicapa, que se distingue por ser un aprendizaje de tipo supervisado y que consiste en la prealimentación de la red.

Finalmente, se obtuvo un software que permite el reconocimiento de las vocales, implementando un sistema de fácil uso que muestra en pantalla la representación visual obtenida por los sensores, una imagen ilustrativa de la posición de la mano con el gesto correspondiente a cada letra, la vocal reconocida y como complemento, el sonido oral de ésta. Debido a que solo se representan cinco letras estáticas, el porcentaje de precisión para cada una de ellas es del 100%, disminuyendo este valor a medida que se aumente el número de letras a ser identificadas. Con base en este resultado, el nivel de error percibido es nulo para cada caso, debido a la validación y calibración empleadas además que los gestos presentados para cada vocal no tienen un nivel de coincidencia entre sí.

Se finaliza lo referente a esta investigación deduciendo que el sistema obtenido es bastante amigable e intuitivo para las personas que presentan discapacidad auditiva y verbal, permitiendo fungir como una base para un intérprete completo que permita una comunicación exacta con el ambiente en que se desarrollan. Se prueba también que la herramienta a base de sensores *Leap Motion* es excelente para aplicaciones que requieren el reconocimiento de gestos, aparte de su facilidad de usabilidad y precisión.

En 2018 Hernández de la Luz et al., haciendo uso de la tecnología *Leap Motion* y tecnología 3D desarrollan un prototipo de intérprete de LSM, cuyo propósito principal es impulsar una base para el desarrollo de un intérprete que represente una herramienta para el apoyo a la mejora en la calidad de vida de las personas con discapacidad auditiva, concediendo el derecho a la comunicación de éstas.

La metodología efectuada se realizó bajo un enfoque cuantitativo, haciendo uso de tres algoritmos, siendo estos la red neuronal *perceptrón*, *máquina de vectores de soporte* que utiliza un enfoque de clasificación binaria y *KNN* que es un algoritmo sencillo que permite la comparación de resultados. La base visual para el reconocimiento en este caso, son la mano, dedos y huesos, así como la información correspondiente a los ángulos y dirección de éstos. Con 2002 registros con alrededor de 81 características cada uno, concernientes a 16 letras del alfabeto de Lengua de Señas Mexicana. Se pudieron determinar las letras que presentaban errores de reconocimiento, por ejemplo, la letra R, así como la comparación de las matrices de confusión, mostrando errores para las letras U y V, precedidas por la F y O.

Concluyendo así que este prototipo presenta un 95% de precisión, lo que implica un alto nivel de aceptación para la interpretación de letras de tipo estático en la LSM. También

es considerable el porcentaje de desacierto de las letras antes mencionadas debido a la posición de los dedos, que se entrecruzan o se posicionan directamente contra la palma de la mano. Las letras que presentan algún tipo de movimiento no son consideradas en esta versión.

Por otra parte, Hernández (2019) presenta el desarrollo de un intérprete de LSM utilizando, al igual que los casos expuestos anteriormente, el periférico *Kinect* y la tecnología *Leap Motion* para realizar el reconocimiento de gestos y posiciones en *tiempo real* del cuerpo y las manos en un contexto específico.

El autor se enfocó en el paradigma cuantitativo y empleando el modelo de trabajo SCRUM con el fin de llevar a cabo un proceso acelerado, pero a la vez permitiendo mantener la calidad del intérprete mediante la captura de los gestos a través de tecnología de reconocimiento 3D para la posición y desplazamiento de los brazos, la captura de imágenes por medio de una cámara web para posteriormente ser procesadas para realizar la comparación y análisis de las características pertinentes. Se obtuvieron 4672 registros totales, de los cuales 480 son correspondientes a gestos visuales, pertenecientes a 146 sujetos que realizaron 32 gestos de señas cada uno para la etapa de aprendizaje con un porcentaje de exactitud de 95%, mientras que para la etapa de prueba se produjeron 3600 registros.

Las pruebas se realizaron 10 veces para conseguir los resultados expresados, donde se pudo cuantificar el mínimo de errores, debidos al color de piel de las personas y la iluminación disponible en el lugar determinado. Existen posiciones peculiares que propiciaron una serie de obstáculos para la identificación de las letras, de acuerdo, a la posición de los dedos (cruces, entrelaces) que se propone como mejora a futuro, así como el equilibrio entre colores para evitar los posibles problemas por iluminación o distinciones en el color de piel. El desarrollo de esta herramienta propició en los sujetos de prueba el entusiasmo para querer practicar e implementar la LSM.

De igual forma en 2019, Mancilla et. Al., elaboraron un estudio sobre el análisis acerca del rendimiento de un sistema clasificador para el reconocimiento de señas de la LSM, con empleo de redes neuronales multicapa y una Máquina de Soporte Vectorial (MSV); en conjunto con las características descriptivas sobre los momentos de Hu, momentos de Zernike, así como histogramas de orientación del gradiente (HOG).

Se consideraron 21 letras para el conjunto de datos con un total de 6300 imágenes, escalando estas a un tamaño específico y segmentándolas, dependiendo del color de piel. El sistema se evaluó mediante una validación cruzada, dividiendo los datos en 70% destinados al entrenamiento y 30% para la prueba.

Los resultados denotaron un mejor desempeño de los momentos de Zernike, con un 98.7% de certeza. Las delimitantes encontradas en este trabajo se denotan por la similitud entre el color de la piel y el fondo u objetos en el ambiente; sin embargo, se puede realizar un ajuste en el rango de detección de color del sistema para su solución.

En 2021, Trujillo y García conformaron el desarrollo de una herramienta de reconocimiento de palabras de la LSM a través de sensores RGB, que son incluidos en dispositivos *Kinect*. Se hizo uso de una red neuronal artificial de tipo *perceptrón* multi capa, que se entrenó con el algoritmo *Backpropagation* y cuyos resultados fueron validados por medio del método *K-Fold Cross Validation*.

Debido al dispositivo utilizado para la obtención de la imagen (cámara RGB y de infrarrojos), se pudieron obtener resultados apropiados para ambientes con todo tipo de iluminación.

Para el CLSM se recabaron 1590 muestras, de 53 palabras que fueron interpretadas por 30 personas pertenecientes a la comunidad signante. Para el reconocimiento, se priorizó la identificación de la mano, y su trayectoria. Considerando el total de 30 muestras para cada palabra, se asignó un valor de *kfold* máximo de 6; a partir de esto se tomaron cinco imágenes de las 30 muestras de cada seña en el CLSM.

De esta forma, se obtuvo un porcentaje de precisión que oscila entre el 90% y 99% de cada iteración *kfold*.

## **2.5 Cálculo Estadístico de Fiabilidad**

Parte importante sobre el análisis de resultados de un estudio, es poder comprobar la fiabilidad de los datos presentados. En este caso, y haciendo énfasis en el uso de descriptores geométricos referentes a los momentos invariantes de Hu, se optó por utilizar la herramienta IBM SPSS que se encarga de examinar de manera estadística y a través del empleo de diversos algoritmos de machine learning, la correlación entre los datos determinados. (IBM, s.f.)

Este software permite canalizar las pruebas de hipótesis *de arriba hacia abajo* para los datos, este es un tipo de procesamiento que se define a partir de la formación de una percepción a partir de un objeto o idea más grande previo a la segregación de dicha idea con información más detallada. Es decir, se va de lo general a lo específico.

## CAPÍTULO 3. METODOLOGÍA DE DESARROLLO

En esta sección se resume la construcción del sistema a partir del uso de las herramientas tecnológicas propias de Python, asociadas al RP en imágenes.

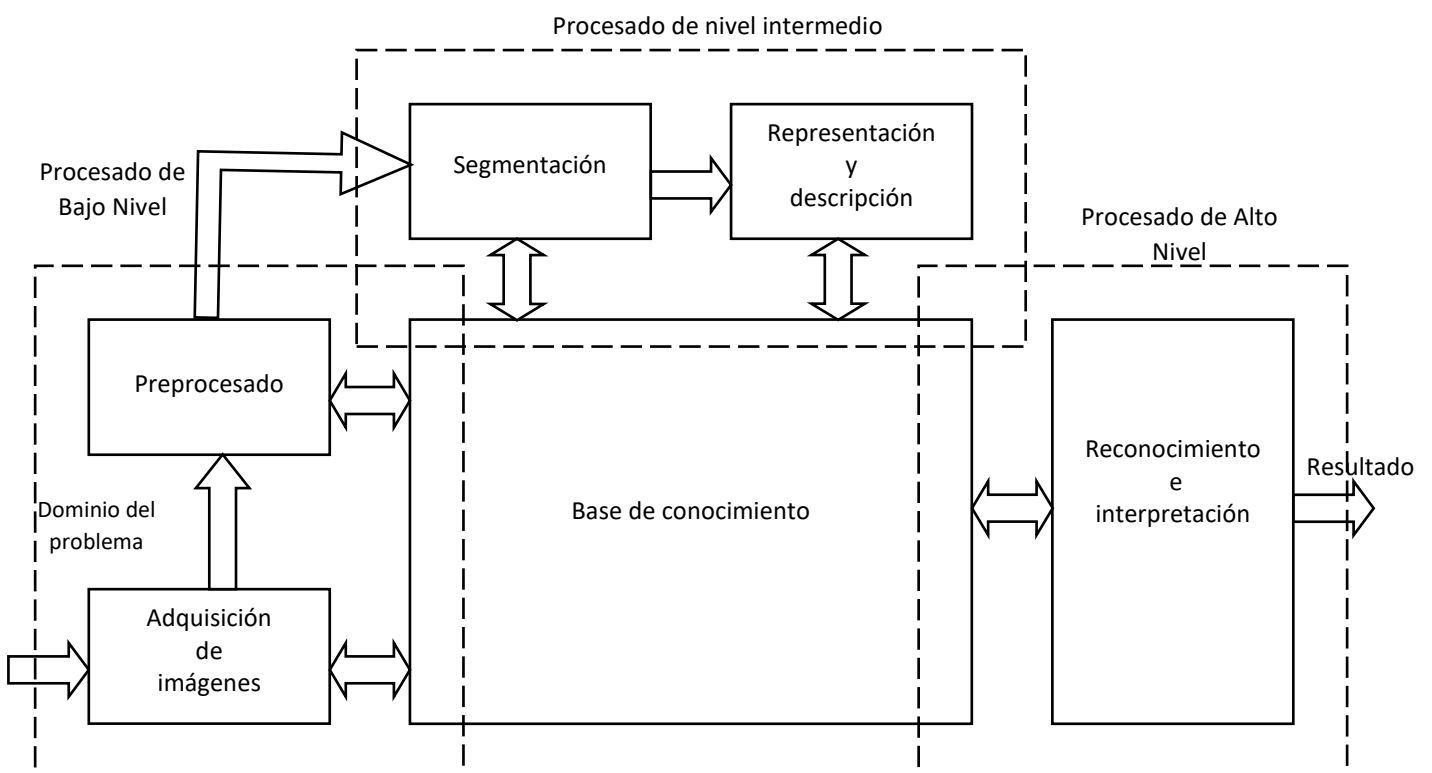
### 3.1 Metodología

La metodología que se utilizó en el desarrollo del sistema de reconocimiento de la LSM basada en aprendizaje supervisado es la descrita por Richard González y E. Woods en su libro “Tratamiento digital de imágenes” (González y Woods, 2018); ya que aborda temas de procesamiento digital de imágenes, pero también propone la utilización de una Base de Conocimientos para realizar reconocimiento e interpretación de patrones.

Esta metodología se divide en tres áreas básicas: 1) Procesado de bajo nivel, 2) Procesado de nivel intermedio, y 3) Procesado de alto nivel. Estas áreas, a su vez, se subdividen en 5 procesos: a) Adquisición de imágenes, b) Pre-procesado, c) Segmentación, d) Representación y descripción y e) Reconocimiento e interpretación (González, 2002). La figura 3.1 ilustra las divisiones y como puede notarse, las etapas empatan con las referidas en la figura 2.2.

**Figura 3.1**

*Diagrama del sistema de visión para el RP de la LSM*



Los bloques de líneas discontinuas se solapan, indicando que no existen límites definidos entre los diferentes procesos. Por ejemplo, la umbralización ó binarización se pueden considerar como una operación de realzado (pre-procesado) o como una herramienta de segmentación, dependiendo de la aplicación concreta.

Cabe mencionar que se utilizó esta metodología con la finalidad de desarrollar un sistema de reconocimiento de patrones, que por sus características y fines (extracción de patrones y clasificación) dista mucho de la metodología empleada para un sistema de información; sin embargo, con apoyo en una fase de análisis fue cómo se limitó el alcance del presente trabajo, así como, bajo una observación visual de la LSM y pruebas de las técnicas y métodos de extracción de los diferentes rasgos fue como se obtuvieron aquellos que fueron de utilidad para lograr una clasificación exitosa. La funcionalidad de esta metodología ha sido probada con éxito en investigaciones y desarrollos tecnológicos que versan sobre extracción de característica, descripción y clasificación.

Por otra parte, este trabajo hace uso de un enfoque cuantitativo debido a que los datos por cada seña son productos de mediciones geométricas como los momentos de Hu, las cuales se representan mediante números (cantidades) y son analizadas a través de métodos estadísticos.

### **3.2 Plataforma de Desarrollo**

Se definió el uso de Python por su similitud en el reconocimiento de patrones y el procesamiento de imágenes con la herramienta Matlab, reconocida herramienta en esta área, que representa el manejo en el tratamiento de imágenes de manera amplia y robusta.

Dicho lenguaje facilita la manipulación de una imagen por medio del uso de librerías para el tratamiento de las imágenes, los cálculos relacionados a los procesamientos que conllevan éste y la gestión de características según convenga al usuario.

La versión precisa para el desarrollo del consiguiente sistema es *Python 3.9.7* debido a la estabilidad que supone respecto con las más recientes versiones de este lenguaje de programación. Asimismo, la disponibilidad que presupone para su empleo en cualquier sistema operativo vigente (Python, 2023).

En relación con las previas propiedades que refieren a Python, y por cuestiones de compatibilidad entre los sistemas operativos del dispositivo de desarrollo y el dispositivo destino (móvil) se optó el llevar a cabo la codificación desde dispositivos con sistema operativo Windows 10, cuyas características se desglosan en la tabla 3.1, de la misma forma que las del dispositivo destino.

**Tabla 3.1**

*Requerimientos recomendables para el desarrollo e implementación del sistema*

	<b>Versión</b>	<b>Procesador</b>	<b>Espacio de almacenamiento</b>	<b>RAM</b>
<b>Dispositivo fuente (desarrollo - computadora)</b>	Windows 10 Home Single Language	2.6 GHz	500 MB	4 GB
<b>Dispositivo destino (implementación - celular)</b>	Android superior a versión 8.0	2.3 GHz	64 GB o superior	4 GB

Entre las características resaltables del uso de Python, se encuentran las librerías responsables del procesamiento de imágenes digitales, facilitadoras en gran parte de la obtención de las funciones propias necesarias para el objetivo deseado, entre las principales: *OpenCV*; su propósito es el tratamiento de las características descriptivas de las imágenes, desde su obtención, filtrado, binarizado, la detección de bordes, y demás propiedades.

Uno más de los atributos a acentuar sobre este lenguaje es su afinidad respecto al desarrollo para dispositivos móviles, a través del framework *Kivy*. La combinación de dicho framework en conjunto con las librerías previamente referidas y la aplicación de la función *MatchShapes()*, parte fundamental de la composición y funcionamiento del sistema, que permite la comparación de la distancia resultante de los momentos de Hu (descriptor de las imágenes preprocesadas), de las imágenes que componen la BC respecto a las imágenes capturadas en *tiempo real*.

Se destaca, además, definiéndose, así como uno de los mayores retos al realizarse este proyecto, la migración hacia el dispositivo destino (sistema operativo Android) a través del uso del constructor responsable de la traducción de Python hacia código propio de Android.

Como primera instancia, se enfatiza en el acceso a los requisitos mínimos del dispositivo para el correcto funcionamiento de la aplicación. De esta manera, una de las bases primordiales es la concesión de permisos (*request\_permissions*) para el uso de la cámara, herramienta principal para la ejecución general del sistema; el almacenamiento interno, así como el externo, para el archivado de la BC y las imágenes tomadas en *tiempo real* (mientras la aplicación se encuentra en ejecución).

Considerada como un fragmento de la parte final de este desarrollo, se efectuó esta construcción dentro de un entorno perteneciente a *Google Colab*, fungiendo como intermediario para la producción del APK para su posterior ejecución en el dispositivo móvil.

### **3.3 Descripción General del Sistema**

De manera general, el sistema se conforma de las siguientes fases:

- I. Adquisición
- II. Preprocesamiento
- III. Segmentación
- IV. Descripción
  - a. Características descriptivas
- V. Clasificación
  - a. Comparación de las características descriptivas
  - b. Obtención de resultados

#### **3.3.1 Adquisición**

La etapa de adquisición se define como la obtención de imágenes en formato digital, para su posterior procesamiento (González y Woods, 2018).

En lo que respecta a la adquisición de imágenes cromáticas, es preponderante mencionar que fueron obtenidas a una distancia aproximada de 30 a 50 centímetros, con la finalidad de capturar de forma íntegra la seña de la mano y una pequeña parte



de la muñeca, recabando imágenes con un tamaño contiguo de 1280 x 1280, tal como se muestra en la tabla 3.2.

En relación con esta etapa, una de las condiciones iniciales fue el determinar el tipo de fondo para las fotografías destinadas a conformar la Base de Conocimientos, disponiéndose el utilizar fondos claros en un principio, con el propósito de permitir la delimitación de la forma de la mano de manera que permitiera la distinción entre fondo y figura; no obstante, al ejecutarse las primeras pruebas del preprocesamiento los resultados obtenidos denotaban cantidades vastas de ruido en las imágenes que impedían la puntualización de la seña. En consecuencia, se optó por la aplicación de un fondo oscuro, destacando la mejora en la obtención del contraste de este frente a la figura de la mano.

Cada una de las fotografías de las señas correspondientes a la dactilología de la LSM fueron adquiridas a través de dispositivos móviles, contemplándose para este caso solo el empleo de la mano derecha. Las características de estas imágenes se listan en la tabla 3.2.

**Tabla 3.2**

*Metadatos de las Imágenes adquiridas para la BC*

<b>Propiedad</b>	<b>Valor</b>
<b>Extensión</b>	JPG
<b>Tamaño en píxeles</b>	1280 x 1280
<b>Resolución</b>	72 ppp
<b>Profundidad en bits</b>	24
<b>Representación del color</b>	RGB

La tabla 3.3 resalta las características del dispositivo móvil empleado para la obtención de las distintas imágenes de las señas de la LSM.

**Tabla 3.3**

*Características del dispositivo móvil: Adquisición de imágenes*

Propiedad	Valor
<b>Sistema operativo</b>	Android 9 Pie con EMUI 9.0.1
<b>Procesador</b>	Kirin 710 GPU Mali G51 MP4
<b>Resolución cámara</b>	Cámara triple de 48MP(f/1.8) + 8MP (f/2.4, gran angular) + 2MP (sensor profundidad, f/1.8)
<b>Memoria</b>	4 GB
<b>Almacenamiento</b>	128 GB

La tabla 3.4 señala las funciones utilizadas para la adquisición de las imágenes de la LSM, empleando el lenguaje de programación Python. La figura 3.2 ilustra el uso de dicha función o funciones.

Así mismo, en la tabla 3.5 es visible la ruta de almacenamiento de la Base de Conocimientos en el proyecto del sistema. La figura 3.3 muestra la imagen original para la letra “V”.

**Tabla 3.4**

*Funciones utilizadas en Python: Adquisición de imágenes*

Nombre	Descripción
<b>imread(nombre_imagen)</b>	Lee una imagen que será tratada a través de un arreglo numpy, con el uso de la librería <i>OPENCV</i> .

**Figura 3.2**

*Código empleado para conformar la BC*

```
imagen = cv.imread('Images/KBImages/A.jpg')
```

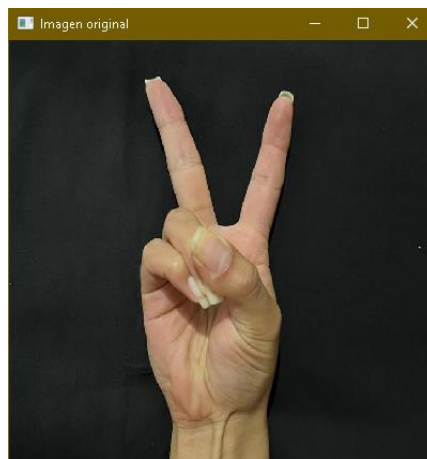
**Tabla 3.5**

*Datos referentes al almacenamiento de la BC*

<b>Nombre</b>	<b>Descripción</b>	<b>Ubicación</b>
<b>KBImages</b>	Imágenes de la BC en formato original (Knowledge Base, KB)	Pdi-python/Images
<b>PreImages</b>	Imágenes de la BC, preprocesadas	Pdi-python/Images
<b>NewImages</b>	Imágenes capturadas en <i>tiempo real</i> , formato original	Pdi-python/Images
<b>RTImages</b>	Imágenes capturadas en <i>tiempo real</i> , preprocesadas	Pdi-python/Images

**Figura 3.3**

*Imagen original de la letra “V”*



### **3.3.2 Preprocesamiento**

El preprocesamiento se conforma de operaciones digitales primitivas, para permitir la reducción de ruido, realce de contraste y estandarización de imágenes en cuanto a elementos como el tamaño y estructura (González y Woods, 2018).

En primera instancia, al tratarse de imágenes capturadas por medio de un dispositivo externo, se eliminó el ruido de las imágenes adquiridas por medio de la función *PyrMeanShiftFiltering()* la cual requiere de la determinación de atributos como el valor de radio espacial, es decir, el valor del rango de los pixeles cercanos al pixel que estemos considerando, y el radio potencial de la cercanía del color de los pixeles vecinos.

Adicionalmente, se aplicó la transformación a *escala de grises* de cada una de las imágenes. La mayoría de las partes del procesamiento digital de las imágenes se llevó a cabo a partir del uso de la librería *OpenCV*.

Posteriormente, y para el correcto manejo de las funciones de procesamiento de la imagen, se *binarizó* cada una de ellas. Lo que concierne a la binarización existen distintos algoritmos de umbrales que provee la librería *OpenCV*, los cuales van desde umbrales simples como lo es *THRESH\_BINARY* el cual aplica un umbral adaptable a una matriz mediante la siguiente fórmula:

$$dst(x, y) = fork\{text\{maxV alue\}\}\{if(src(x, y) > T(x, y))\}\{0\}\{else\} \quad (1)$$

Dónde:

- Src es la fuente de la imagen en un canal de 8 bits
- Dst es la imagen de destino
- MaxValue es un valor distinto de 0 asignado a los pixeles que cumplen la condición

Hasta umbrales más especializados como *THRESH\_OTSU* que utiliza técnicas estadísticas como la varianza, que en lugar de elegir un valor arbitrariamente como umbral, Otsu lo determina automáticamente a partir de los valores pico del histograma de la imagen.

El algoritmo de Otsu encuentra un valor(t) que minimice la varianza ponderada, es decir determina el valor del umbral a partir del valor que esté entre dos picos tales que sus varianzas sean mínimas. La fórmula que utiliza este algoritmo es la siguiente:

$$\sigma^2w(t) = q_1(t)\sigma^2_1(t) + q_2(t)\sigma^2_2(t) \quad (2)$$

Donde:

- $q_1$  es el primer valor pico del histograma de la imagen;
- $q_2$  es el segundo valor pico del histograma de la imagen;

-t el valor del umbral calculado

Asimismo, existen otros algoritmos de umbrales, sin embargo, los antes mencionados son los más significativos para este proyecto, ya que arrojaron preminentes resultados de visualización en las imágenes y por consiguiente los utilizados en este trabajo de investigación.

Finalmente, para perfeccionar la detección de bordes, se aplicó el operador Sobel el cual adiciona ciertos valores en la máscara que aproximan a un suavizamiento Gaussiano. Este operador fue elegido sobre el operador Prewitt ya que Sobel presentó mejor detección de bordes diagonales.

En las pruebas realizadas con las primeras imágenes fue necesario aplicar *filtros* de mejora de la imagen para el contraste y reducción de ruido. De aquí, que las fases consideradas fueron *eliminación de ruido, conversión a escala de gris, binarización y filtrado*. La tabla 3.6 muestra las funciones de Python correspondientes, y la implementación de estas en el código se muestra en la figura 3.4. Las figuras 3.5 a 3.8 ilustran la aplicación de esta etapa.

**Tabla 3.6**

*Funciones utilizadas en Python: Preprocesamiento de imágenes*

Nombre	Descripción
<b>PyrMeanShiftFiltering((imagen, radio_espacial, radio_color)</b>	Elimina el ruido de la imagen, donde el primer parámetro es la imagen origen, el segundo parámetro es el valor de cercanía espacial de los pixeles vecinos y el tercero es el valor del radio de la ventana de color.
<b>cvtColor(imagen, código_conversión)</b>	Transforma la imagen a escala de grises, en este caso, a través de la función <i>COLOR_BGR2GRAY</i> (código de conversión).
<b>Threshold(imagen, umbral, valor_máximo, algoritmo_umbral)</b>	Binariza la imagen, siendo el primer valor de la función la imagen transformada a escala de grises, el segundo argumento contiene el umbral para hacer la

	clasificación de los valores del píxel, el tercero es el valor máximo que se le asigna al píxel en dado caso que se supere el valor del umbral previamente establecido, y el cuarto parámetro es el tipo de umbral que podemos aplicar a la imagen, (en este caso se aplicó <i>THRESH_BINARY   THRESH_OTSU</i> )
<b>sobel()</b>	Detecta los bordes horizontales, verticales y diagonales.

### Figura 3.4

*Código del preprocesamiento de las imágenes.*

```
# Pre-procesamiento de imagen

# Eliminación de ruido
im_ruido = cv.pyrMeanShiftFiltering(imagen,sp=30,sr=50)

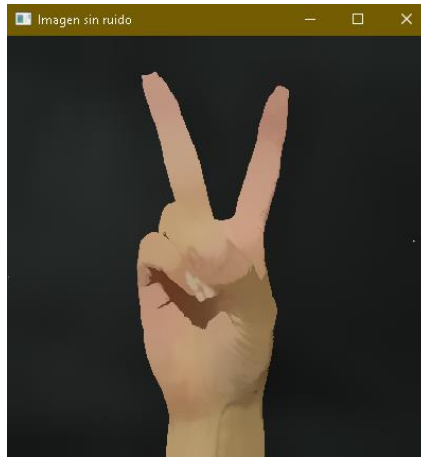
# Escala de grises
im_gris = cv.cvtColor(im_ruido, cv.COLOR_BGR2GRAY)

# Binarización | Binarización de Otsu
ret, im_binaria = cv.threshold(im_gris, 0, 255, cv.THRESH_BINARY |
cv.THRESH_OTSU)

# Detección de bordes
img_filtro1 = filters.sobel(im_binaria)
```

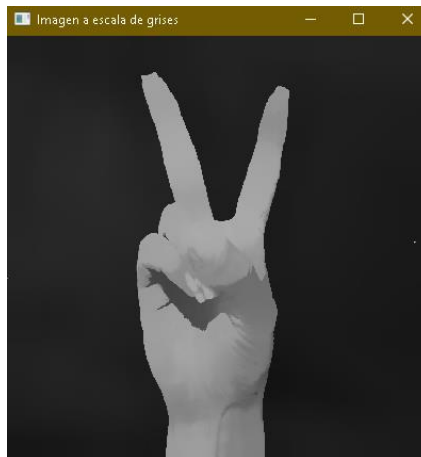
**Figura 3.5**

*Eliminación de ruido en la imagen para la letra "V"*



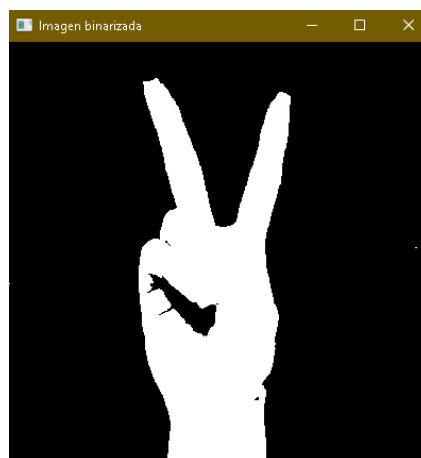
**Figura 3.6**

*Transformación a escala de grises de la imagen para la letra "V"*



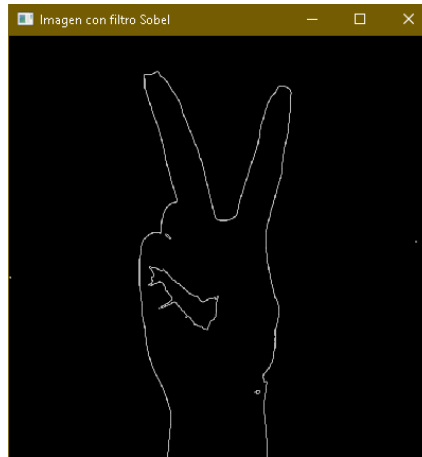
**Figura 3.7**

*Binarización de Otsu aplicada a la imagen para la letra "V"*



### Figura 3.8

*Detección de bordes de la imagen para la letra “V”*



#### 3.3.3 Segmentación

La segmentación se considera parte del proceso de reconocimiento de patrones; Duda, Hart y Stork (2000) definen a la segmentación como la capacidad de separar dos o más figuras entre sí, o del fondo, con el propósito de obtener de manera fácil los descriptores que permitan su posterior clasificación.

En consecuencia, la finalidad de esta etapa se define como la facultad para segregar la figura de la mano derecha frente al fondo determinado, incluso para señas con aberturas como en el caso de la letra “O”. Esta acción fue uno de los desafíos cruciales, por lo que funciones como la asociada a la dilatación, encargada de la expansión de píxeles en los bordes de la imagen; y llenado de huecos fueron necesarias.

El tipo de segmentación utilizada de acuerdo con Jähne (2002) se establece como segmentación basada en los píxeles debido a la simpleza que comprende la distinción del fondo versus el borde de la figura, dado a la uniformidad de los píxeles en el área de interés (seña con píxeles en blanco) versus el fondo (píxeles en negro).

En la tabla 3.7 se muestran las funciones empleadas para cumplir el objetivo de la diferenciación de la figura de la seña contra el fondo, así como las especificaciones correspondientes. La figura 3.9 muestra la implementación de estas funciones en el código. La aplicación de esta etapa en las imágenes puede observarse en las figuras 3.10 a 3.15.



**Tabla 3.7***Funciones utilizadas en Python: Segmentación de imágenes*

Nombre	Descripción
<b>threshold(imagen, umbral, valor_máximo, algoritmo_umbral)</b>	Convierte la imagen a formato binario, para facilitar la manipulación y tratamiento de ésta a través de la función <i>THRESH_BINARY</i> (técnica o algoritmo del umbral). Asigna píxeles en relación con el umbral definido ( <i>threshold</i> ).
<b>dilate (imagen, kernel, iteraciones)</b>	Añade píxeles a los límites establecidos en la imagen determinada, generalmente, píxeles blancos. Es decir, las áreas de píxeles se expanden respecto a su tamaño original haciendo que los agujeros existentes en estas regiones se vuelvan más pequeños.
<b>GetStructuringElement()</b>	Mejor conocido como elemento estructurante o <i>kernel</i> que decide la naturaleza de la operación, en este caso uno de tipo rectangular de 3X3.
<b>floodFill(imagen, máscara, valor_alto, valor_bajo)</b>	Conecta un píxel con sus vecinos cercanos ADYACENTES, que cumplan con un umbral de diferencia cercano al de dicho píxel. Es decir, comprende el llenado de huecos, lo que permite rellenar la figura deseada haciendo correspondencia a la operación morfológica de cierre. Esta función emplea un tipo de adyacencia 8-conexa.
<b>connectedComponents(imagen, etiquetas, conectividad, tipo)</b>	Etiqueta la imagen a partir de la extracción de una región de ésta, por medio de componentes conectados por alguna característica similar, como el color del píxel.

### Figura 3.9

Código de la segmentación de imágenes

```
# Binarización | Binarización de Otsu - Segmentación
ret, im_binaria = cv.threshold(im_gris, 0, 255, cv.THRESH_BINARY |
cv.THRESH_OTSU)

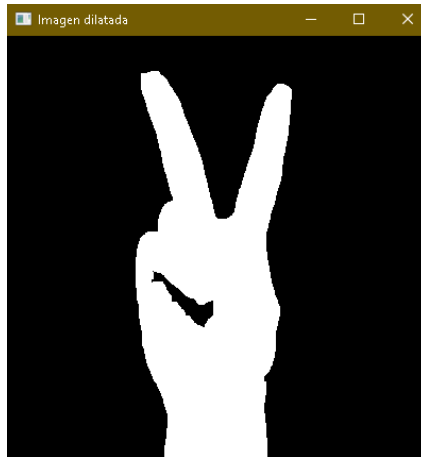
# Operaciones morfologicas - Segmentación
kernel = cv.getStructuringElement(cv.MORPH_RECT, (3, 3))
apertura = cv.morphologyEx(im_binaria, cv.MORPH_OPEN, kernel=kernel,
iterations=2)
dilatacion = cv.dilate(apertura, kernel, iterations=3)

# Umbralización - Segmentación
th, im_umbral = cv.threshold(apertura, 80, 150, cv.THRESH_BINARY_INV)
# Copia de la imagen umbralizada - Segmentación
im_umbral_copia = im_umbral.copy()
# Mascara para rellenar con 2 pixeles mayores a la imagen - Segmentación
[Relleno]
a, l = im_umbral.shape[:2]
mascara = np.zeros((a+2, l+2), np.uint8)

# Rellenando desde el punto (0, 0) al 255
cv.floodFill(im_umbral_copia, mascara, (0,0), 255);
# Invertido del relleno de la imagen - Segmentación [Relleno invertido]
im_seg_inv = cv.bitwise_not(im_umbral_copia)
# Combinación de imagenes para obtener imagen rellenada - Segmentación
im_combinada = im_umbral | im_seg_inv
# Binarización de otsu - Segmentación
ret, im_binaria2 = cv.threshold(im_combinada, 0, 255, cv.THRESH_BINARY |
cv.THRESH_OTSU)
```

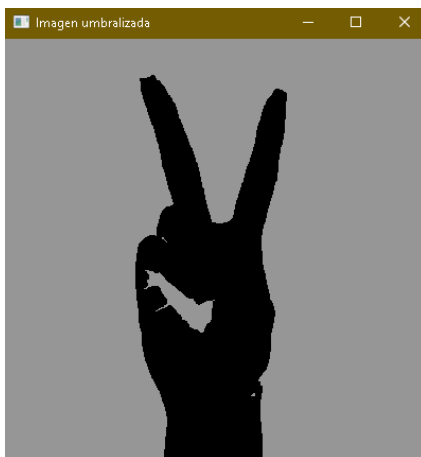
**Figura 3.10**

*Dilatación de la imagen para la letra “V”*



**Figura 3.11**

*Umbralización de la imagen para la letra “V”*



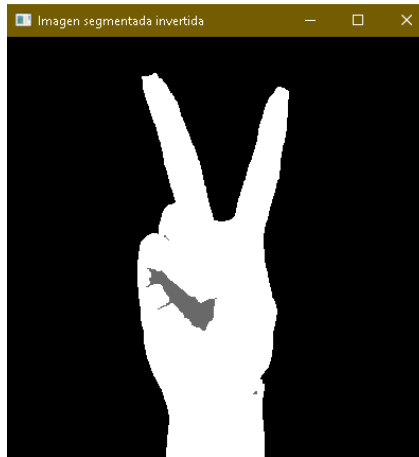
**Figura 3.12**

*Segmentación de la imagen para la letra “V”*



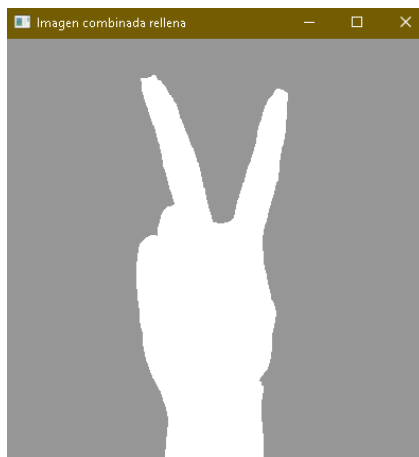
**Figura 3.13**

*Segmentación invertida de la imagen para la letra "V"*



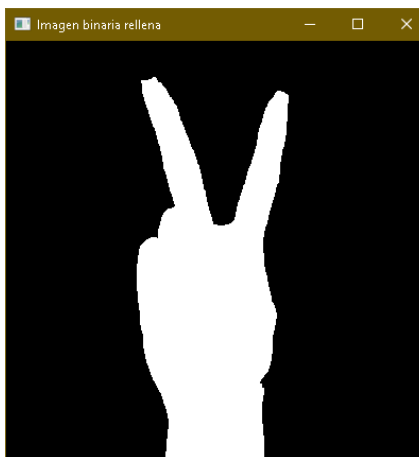
**Figura 3.14**

*Relleno de la imagen para la letra "V"*



**Figura 3.15**

*Binarización de la imagen rellena para la letra "V"*



### **3.3.4 Descripción**

La descripción de imágenes corresponde a la detección de características cuantitativas con el propósito de encontrar datos o elementos en la imagen, región o límites de esta, para su uso posterior (González y Woods, 2018).

Tras el breve estudio de algunos de los descriptores respecto al tipo de imágenes que se analizan, se consideraron los siguientes.

#### **3.3.4.1 Longitud de la Figura/Seña.**

La longitud o medida de una figura suele definir el área o perímetro de esta, tratándose de la cantidad de píxeles que se encuentren delimitados por un contorno o formando parte de este, respectivamente (Universidad Nacional de San Juan, s.f.).

#### **3.3.4.2 Invariantes Projectivos.**

Agrupación de puntos distinguibles en la imagen con relación a la perspectiva, permiten la identificación del objeto sin importar el punto de vista desde el que se “observe” éste (Duda y Hart, 1973).

#### **3.3.4.3 Regiones.**

Técnica de segmentación que permite subdividir la imagen en las áreas de interés respecto al resto, mediante la agrupación de píxeles vecinos que presentan características similares. (La Serna y Román, 2009)

#### **3.3.4.4 Momentos de Hu.**

Son los momentos de las imágenes que se calculan a partir de las distintas transformaciones que pueden ser aplicadas a éstas, como son la rotación, traslación, escalación y reflexión y son siete momentos de acuerdo con González y Woods (2018).

En la tabla 3.8 se muestra las características de comparación que permitieron la elección de los descriptores.

**Tabla 3.8**

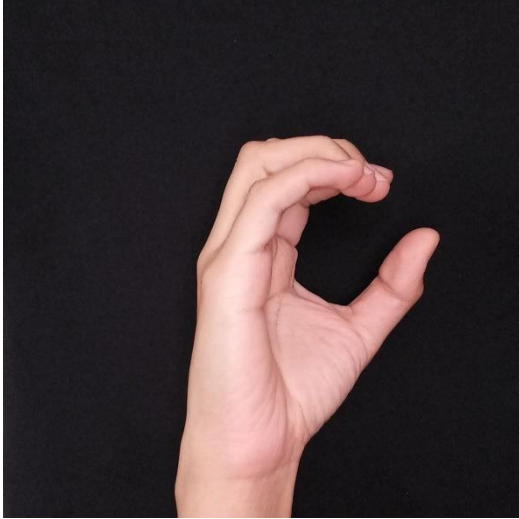
*Comparación de los posibles descriptores de las imágenes*

Nombre	Utilizado/Descartado	Justificación
<b>Invariantes proyectivos</b>	X	No representa algún beneficio para el objetivo del sistema ya que, al poder reconocer el objeto desde cualquier perspectiva, se obtienen resultados no útiles cuando se encuentren configuraciones de las señas que sean semejantes, vistas desde distintos ángulos. Por ejemplo, para las letras “M” y “N”, considerándose un punto de visualización por encima de la mano, o las letras “T” y “S”, bajo esta misma premisa.
<b>Longitud de la mano/seña</b>	X	En este caso no es posible implementar este tipo de atributo puesto que, al configurarse cada seña con las manos, es de conocimiento general que no todas poseen tamaños determinados para su clasificación.  Las figuras 3.16 y 3.17 ilustran lo referido para este descriptor.
<b>Momentos de Hu</b>	✓	Se recabaron los momentos de Hu de las imágenes obtenidas para cada una de las señas estáticas de la LSM pertenecientes a la BC para una muestra de cinco manos derechas distintas, determinando como imagen original o central a aquella con un ángulo de 0° de inclinación. La figura 3.19 muestra el código de implementación de estos.  La tabla 3.9 ilustra los momentos de Hu, donde se pueden observar datos concisos sobre cada una de las imágenes a las que se aplicaron, estos datos facilitan la comparación entre señas.
<b>Regiones</b>	X	No resultó fiable aplicar este método ya que letras como la “M” y “N” presentaban cantidades análogas de regiones, haciendo inasequible su

reconocimiento y diferenciación de estas. La figura 3.18 puntualiza este señalamiento.

**Figura 3.16**

*Seña letra "C" (mano derecha #1)*



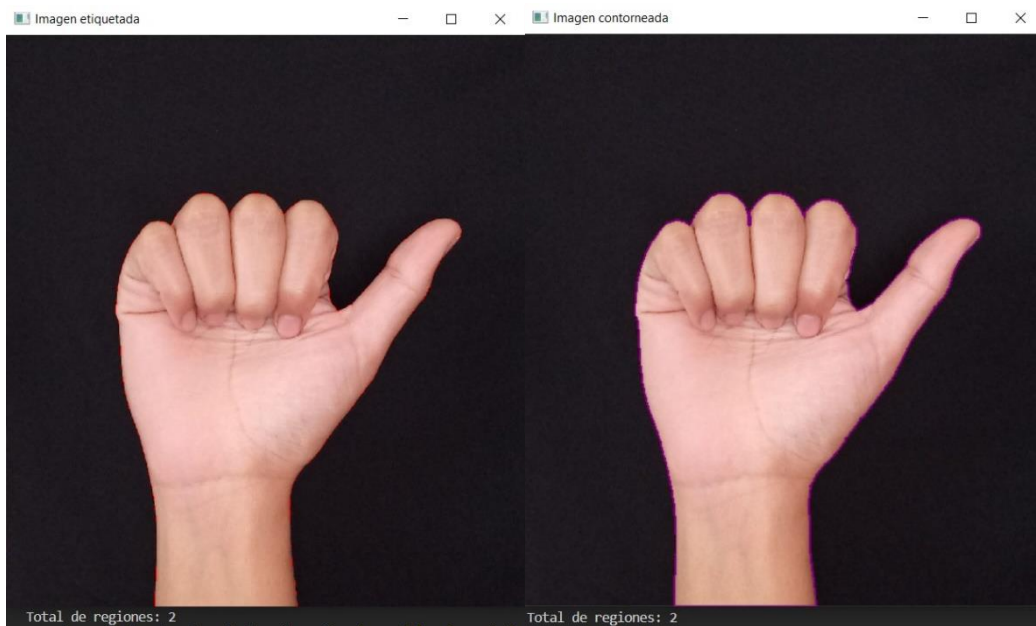
**Figura 3.17**

*Seña letra "C" (mano derecha #2)*



**Figura 3.18**

*Seña letra "A", regiones*



### Figura 3.19

*Código de la obtención de los descriptores (momentos de Hu)*

```
def momentos(imagen):
    mostrarMomentos = True
    moments = cv.moments(imagen)
    huMoments = cv.HuMoments(moments)




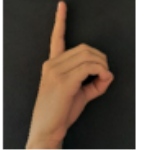




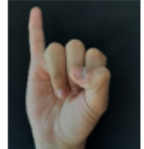

    for i in range(0,7):
        if mostrarMomentos:
            print((-1*copysign(1.0,\
                huMoments[i])*log10(abs(huMoments[i]))),\
                end='\n')
        else:
            print("{:.5f}".format(huMoments[i]),end=' ')
```








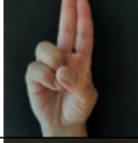
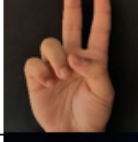
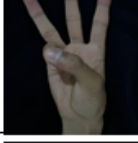
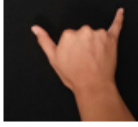
Como podemos observar, los siete momentos de Hu se determinan tras aplicar la operación de logaritmo, que se desglosan en las ecuaciones 5 y 6.



**Tabla 3.9**

*Momentos de Hu para cada una de las letras estáticas de la LSM*

Letra	ID	H0	H1	H2	H3	H4	H5	H6	Ilustración
<b>A</b>	1	3.1011198	7.08365042	9.77567816	11.1576872	-21.6351497	-14.8578377	22.2818101	
	2	3.08773783	6.92709842	9.92019663	11.313007	-21.96003	-15.3431002	22.3714307	
	3	3.12558474	7.18162345	9.90455212	11.3003803	-22.1950059	-14.9573576	-21.9683551	
	4	3.0932614	7.16520722	9.64585249	10.9963297	-21.4608034	-14.5852319	-21.4753104	
	5	3.13592463	7.34339877	10.0352411	11.4479829	-22.6300451	-15.1917963	-22.2202231	
<b>B</b>	1	3.03738532	6.3559304	11.1697695	11.673549	23.1022889	14.8657041	-23.8420804	
	2	3.02022724	6.30376887	11.3281084	11.9455235	23.5842682	15.1036785	-24.6090471	
	3	3.0389145	6.35604008	12.1412772	12.7436076	25.1918401	15.957606	25.975975	
	4	3.06226367	6.45457009	11.0198468	11.5531025	22.8395787	14.781481	-25.400333	
	5	3.00979708	6.26033731	10.7474757	11.2089511	22.1890051	14.3425583	-23.2239879	
<b>C</b>	1	3.00618793	6.46049992	10.2990685	11.8525283	-22.9324211	15.1066824	-23.7926534	
	2	2.95361897	6.19528546	10.2863368	12.9173797	-24.9132001	-16.0262573	24.5578643	
	3	3.0009272	6.45103638	10.5419855	12.6340099	-24.2234362	15.9317095	25.3136432	
	4	2.99503752	6.41574879	10.8563232	11.2480782	-22.304544	14.5117206	23.1558176	
	5	2.9651587	6.27674069	10.7916141	10.9125905	22.0582459	14.3795746	21.8297127	
<b>D</b>	1	3.03495639	6.52829044	9.64390529	10.9012361	22.7243132	14.9834877	-21.1739789	
	2	2.98371744	6.2235198	9.99737948	10.8405947	21.3341201	14.0919861	-21.5279731	
	3	3.00937893	6.36914718	9.79550714	11.040791	22.0832708	14.7042521	-21.4715479	
	4	3.02982591	6.54905562	9.58880454	10.9316667	-21.4205504	-14.2160597	21.2850793	
	5	3.02487819	6.41639573	9.88000584	11.0476498	21.7807945	14.5754712	-21.5856381	
<b>E</b>	1	3.10725851	6.71352012	10.6327821	11.3761029	22.3879059	14.775027	-23.1191396	
	2	3.08527391	6.60627881	10.5889572	11.2244993	22.1322645	14.5368027	23.2922401	
	3	3.15781918	7.20135177	10.8136534	12.5234608	24.2228182	16.3761036	-24.6313358	
	4	3.17015087	7.25094655	11.2059352	12.814715	25.374245	17.3084078	-24.8430811	
	5	3.12348846	6.83220511	10.563788	11.6547644	22.851575	15.3328663	-23.0036271	
<b>F</b>	1	3.05020553	6.43361849	10.7033294	11.3689061	22.4140972	14.725625	23.0990381	
	2	3.00398487	6.27355326	10.3792152	10.9296992	21.5856028	14.1350022	22.673116	
	3	3.06021293	6.47713366	10.6096185	11.3614826	22.3639475	14.7584684	22.9096856	
	4	3.09515682	6.65235759	10.6437567	11.1767918	22.0870694	14.5440901	24.4995337	
	5	3.0322482	6.35905912	10.3412607	10.9876215	21.6811742	14.3262091	22.1027997	
<b>G</b>	1	3.07239504	6.65367905	10.794096	10.7898056	21.6118794	14.1211224	-22.0255716	
	2	2.98932909	6.25055685	10.2781415	11.2913783	22.6063198	14.8889378	22.0959084	
	3	3.08053848	6.70040013	10.4734879	11.8625622	23.2298196	-15.7147898	-23.141337	
	4	3.05255981	6.8127506	10.3467546	10.5205853	-22.9719263	13.9385166	20.9542752	
	5	3.0462312	6.75093757	10.6766665	10.8154911	-21.5819945	14.3837414	-22.0850029	
<b>H</b>	1	3.06822493	6.58556335	10.5162178	10.7635834	21.4038682	14.0713307	22.7797794	
	2	2.95301086	6.14813389	9.80354107	10.7795882	-21.5971878	-14.307585	21.0913224	
	3	3.03704787	6.46653937	10.0418791	11.1294269	22.2760586	-14.8667172	-21.7321303	
	4	3.06262911	6.7575465	10.0730357	10.6791188	21.1772091	14.0828592	21.2385273	
	5	3.04015234	6.49532967	10.4833388	11.0072534	21.9522851	14.2610146	21.862965	
<b>I</b>	1	3.10451655	6.82350556	10.1369181	11.5840649	-22.4454988	-15.0269839	-23.6262838	
	2	3.069926	6.56683984	10.2237261	11.3773117	-23.0921804	-15.261633	-22.1810761	
	3	3.09464825	6.81907854	10.0531015	11.6293708	-22.573791	-15.0445443	22.6817308	
	4	3.12398417	7.05471054	10.0849347	11.1357852	21.8253264	15.8307271	22.0036003	
	5	3.09177957	6.65409359	10.5263709	11.6112891	23.0530648	-15.5004192	22.7230846	
<b>L</b>	1	3.05229916	6.85616258	11.3528517	10.5288019	-21.4965816	13.9895806	-21.9360402	
	2	2.9968303	6.43401146	10.7997948	10.748335	21.7389513	14.2393714	21.6223476	
	3	3.10542123	7.32937644	11.7937013	11.0469293	-22.9762618	15.2140436	22.4891442	
	4	3.0580174	6.99571146	10.2427206	10.1641757	20.6627389	13.6691267	-20.4321013	
	5	3.06352111	6.91845806	10.7372915	10.767966	-21.5302861	14.4235931	22.2006086	

<b>M</b>	1	3.13045075	6.93686906	10.3217495	11.4246148	22.3069034	14.9095269	22.9910387	
	2	3.08891612	6.65092855	10.5534082	11.2178531	22.1434742	14.5875299	-22.4905717	
	3	3.1530738	7.17100139	10.9977045	11.6808202	23.1042239	15.3468802	-23.2666688	
	4	3.14178356	7.14636201	10.1471326	11.3978504	22.1826331	15.0042221	22.8000368	
	5	3.10682473	6.7564608	10.9221986	11.9140428	23.5169062	-16.1263548	-23.4531185	
<b>N</b>	1	3.10767134	6.85868249	10.0867948	10.9194274	21.4397304	14.3600481	-21.9817934	
	2	3.03355459	6.42144386	10.2428365	10.8117483	21.383195	14.0693787	-21.7066344	
	3	3.1285262	6.99561166	10.1876543	11.2661942	22.0053589	14.7915505	22.6236892	
	4	3.1493289	7.14556059	10.3722879	11.5889023	22.7380417	15.4685226	22.7033659	
	5	3.0777511	6.57855216	10.8432128	11.4903145	22.796138	15.1605599	-22.8196865	
<b>O</b>	1	3.067984	6.52287408	10.2417218	10.8878852	21.4559771	14.1504428	22.3642197	
	2	3.03290515	6.35382855	10.5303563	11.0161318	21.7894912	14.1961315	-23.4266576	
	3	3.10689825	6.74297552	10.378387	11.2264684	22.0314744	14.600544	22.992891	
	4	3.10546997	6.72414441	10.3426694	11.1389018	21.8858968	14.5208968	22.6546366	
	5	3.08385776	6.61109473	10.2637266	10.9643967	21.6177187	14.3270116	21.969193	
<b>P</b>	1	3.01252861	6.44543815	10.0064941	11.9368639	-23.4194474	15.2135609	22.9302431	
	2	2.95432432	6.16237978	10.2774418	12.6712426	24.6229712	16.1900879	24.171128	
	3	3.03091475	6.50228696	10.470904	10.6703506	21.3493429	14.0222111	21.4438445	
	4	3.02427117	6.50125849	10.6327483	10.8552484	-21.6258476	14.120061	22.068341	
	5	2.93722311	6.16261698	11.1548192	10.4355243	-21.278464	13.5277176	-21.5829514	
<b>R</b>	1	3.0321106	6.38243713	10.1160498	10.4157022	20.6826388	13.620809	-21.8377209	
	2	2.95728699	6.11386999	10.0035183	10.2378109	20.3585075	13.2975172	22.2740411	
	3	3.01338536	6.30771291	9.9258138	10.3011851	20.4151533	13.4639387	-21.7478379	
	4	3.01003713	6.32081016	9.87670514	10.4316114	20.607826	13.7388929	-21.0933144	
	5	2.94422546	6.08819111	9.74818942	9.90516223	19.7320824	12.9510894	21.2063376	
<b>S</b>	1	3.14228031	6.98219167	10.6536736	11.7369776	22.9415694	15.2505334	-23.6218474	
	2	3.14308407	7.01766499	10.6185194	11.586745	22.6902868	15.1097692	-23.8787785	
	3	3.11604589	6.77437235	10.6455686	11.5873044	22.7437754	15.0261519	23.090611	
	4	3.16271128	7.18351284	10.8980208	12.1859043	23.770872	15.844205	24.1006315	
	5	3.1190997	6.79111046	10.5807477	11.4928624	22.5485679	14.9710342	-23.0691896	
<b>T</b>	1	3.1281749	6.85574117	10.6454038	11.746058	22.9732449	15.2178407	23.3768528	
	2	3.09929329	6.67424019	10.5821374	11.320549	22.2742037	14.6620235	23.2594721	
	3	3.12427571	6.87016723	10.4175466	11.7212218	-22.7933818	-15.1583761	23.7386764	
	4	3.12101532	6.8346656	10.4474222	11.7749217	-23.6118577	-16.0534229	22.8939103	
	5	3.1095911	6.73119413	10.6620817	11.8621611	23.3176761	15.4999849	23.2390056	
<b>U</b>	1	3.03714987	6.41668095	10.1171649	10.4871055	20.7892541	13.7174338	22.8941441	
	2	2.99067139	6.22373536	10.0426814	10.4445908	20.6908291	13.6032168	-21.6502338	
	3	3.01705491	6.33212422	9.93072056	10.3731069	20.5259178	13.5583299	-21.7174146	
	4	3.03071541	6.40248543	9.83009706	10.2626341	20.3090184	13.473983	-22.3406501	
	5	3.00257482	6.26185259	10.0641348	10.340399	20.5429268	13.4715357	22.002869	
<b>V</b>	1	3.03130958	6.45915884	10.2843628	10.4373242	20.8024992	13.7161636	21.6503883	
	2	2.96276695	6.24469932	11.2183792	10.1535257	20.9202196	13.2861618	21.0934284	
	3	2.96332381	6.20479155	11.5080913	10.4818743	21.5877231	13.5951424	21.6759146	
	4	3.02813612	6.42989787	9.92103801	10.2263281	20.3000473	13.4527707	-22.1893822	
	5	2.9694628	6.26569427	10.9358538	10.1424035	20.6955174	13.2812208	21.2840297	
<b>W</b>	1	2.99791794	6.42807694	10.6110743	10.6175634	-21.2563332	13.8331812	21.7182229	
	2	2.97852135	6.31121324	10.4938761	10.9151856	-22.2362864	14.1388477	21.6327971	
	3	3.00241406	6.35478249	10.6020986	11.0988702	21.9542539	15.0086322	22.7751112	
	4	2.97860999	6.28433127	10.5574109	10.8859267	21.7880943	14.1710265	21.731769	
	5	2.93875567	6.21129551	10.3164776	10.6407056	-21.1198949	13.746371	22.3997179	
<b>Y</b>	1	3.07573617	6.97778681	9.69539132	12.8137082	-24.1600669	17.1512785	24.2994822	
	2	3.04299566	6.79226938	9.58525965	12.1506904	-23.1160285	17.281072	23.2397109	
	3	3.09906572	7.43430724	9.64737013	12.5658878	23.7415269	-18.8205253	-23.9550298	
	4	3.07700716	8.00562477	9.47635661	11.114543	-21.4167831	15.4334726	-22.1658106	
	5	3.0746499	7.92515639	9.46517724	11.4394691	-22.2318235	15.4087688	21.9426766	

### 3.3.5 Clasificación

La clasificación se define como la capacidad de asignar una etiqueta de pertenencia a alguna clase, a un patrón de entrada (Narasimha y Susheela, 2015).

Por otra parte, un clasificador es el algoritmo que produce como resultado la etiqueta perteneciente a una clase determinada a partir de las colecciones de valores sustraídas de los datos de entrada (Kuncheva y Whitaker, 2014).

Existen, de manera general, dos tipos de clasificadores:

**Paramétricos:** Corresponden a las distribuciones de probabilidad, donde, a partir de la información destinada al entrenamiento, se pueden estimar los parámetros a ser empleados. Se incluyen los clasificadores *discriminantes lineales* y *cuadráticos*.

**No paramétricos:** El clasificador de K-vecinos más cercanos (KNN) es un digno representante de este tipo de clasificador, en este caso, se asigna X (punto de datos) a la clase con mayor frecuencia entre sus K vecinos menos distantes.

(Kuncheva y Whitaker, 2014)

En este caso, el clasificador destinado fue de tipo no paramétrico, es decir, el algoritmo de KNN.

#### 3.3.5.1 Algoritmo de KKN.

De manera general, se define como un algoritmo de enfoque supervisado que, a través del uso de la proximidad, realiza la clasificación sobre un conjunto de datos, encontrando puntos similares entre un dato y otro.

La clasificación se realiza al adjudicar una etiqueta de clase, aquella que se presenta con más frecuencia, sobre un punto de datos específico.

Este algoritmo se considera parte de una familia de modelos llamados de “aprendizaje perezoso”, puesto que solo acumula un conjunto de datos de entrenamiento en lugar de realizar por toda la etapa de entrenamiento. Así mismo, el cálculo se ejecuta a la hora de realizar la clasificación.

En conclusión, su propósito es identificar los vecinos más cercanos con respecto a un punto determinado, de manera que sea posible asignar una etiqueta de clase a dicho punto.

Para calcular KNN, es necesario contar con las métricas de distancia, que sirven para determinar la cercanía de puntos con respecto al punto de consulta, permiten determinar los límites de decisión, dividiendo a los puntos de consulta en regiones distintas.

La medida de distancia elegida para este trabajo de investigación es la distancia Manhattan ( $p=1$ ).

### **3.3.5.2 Distancia Manhattan.**

Es una de las métricas de distancia reconocidas, esta mide el valor absoluto entre dos puntos, y se define por la ecuación 3. Debido a que usualmente se visualiza como una cuadrícula, se conoce también como distancia de taxi o de cuadra de la ciudad.

$$\text{Distancia Manhattan} = d(x, y) = \left( \sum_{i=1}^m |x_i - y_i| \right) \quad (3)$$

(IBM, s.f.)

La comparación de las características descriptivas de las imágenes en este caso, se basa en el cálculo de las distancias entre los momentos de Hu de cada una de las figuras examinadas a través de la suma de las diferencias de cada una de las invariantes de Hu.

Para poder ejecutar el reconocimiento de las imágenes también denominados *frames*, obtenidos a partir de la captura de estos a través de video en *tiempo real*, se hace uso de la función *matchShapes*, que a partir de los momentos de Hu, dispone de la distancia entre dos figuras o imágenes de características relativamente similares para poder obtener un número decimal de salida que describe la relación entre ambos sujetos de comparación. El análisis de *matchShapes* (4) declara que, entre más cercano a cero se encuentre el número resultante, más parecidas son las figuras examinadas.

$$\text{matchShapes}(\text{Imagen1}, \text{Imagen2}, \text{método}, \text{parámetro}) \quad (4)$$

Donde:

Imagen1 = es el primer objeto o imagen para comparar;

Imagen2 = es el segundo objeto para la comparación versus el primer objeto;

método = procedimiento a ser aplicado a los objetos en cuestión, en este caso, *CONTOURS\_MATCH\_I2*, encargado del análisis de contornos entre ambos objetos o imágenes;

parámetro = número de tipo *double* que especifica el parámetro a utilizar para la comparación, siendo en su mayoría de casos equivalente a cero.

El funcionamiento de *matchShapes* se define a partir de la comparación de dos objetos siendo éstos imágenes, por medio de la obtención de los momentos de Hu para cada una de estas, y la diferencia entre dichos momentos con base en las ecuaciones 5 y 6 a continuación. Estas ecuaciones se encargan de obtener el producto entre el signo designado por el vector perteneciente al momento de Hu (del 1 hasta el 7) y el logaritmo para este mismo. Se determina el uso de la operación de logaritmo debido a que, para la mayoría de las ocasiones se encuentran momentos de Hu con cifras exponenciales, permitiendo que a través del resultado logarítmico se tengan cifras de mejor legibilidad para su uso posterior, como lo es en este caso, de la distancia entre las imágenes.

$$m_i^A = \text{sign}(h_i^A) \cdot \log(h_i^A) \quad (5)$$

$$m_i^B = \text{sign}(h_i^B) \cdot \log(h_i^B) \quad (6)$$

Donde:

A = objeto o imagen #1;

B = objeto o imagen #2;

$h_i^A$  = momentos de Hu para A;

$h_i^B$  = momentos de Hu para B;

$i = [1:7]$

*CONTOURS\_MATCH\_I2* fue el método elegido debido a que representa tiempos de ejecución e implementación óptimos y de “menor costo” con respecto a los otros dos métodos disponibles para la función, que únicamente difieren en el tipo de operación a ser aplicada, siendo el inverso a la diferencia entre los momentos de Hu y el máximo relativo entre diferencias, respectivamente.

Se resume el método empleado como la suma del valor absoluto entre las diferencias de los momentos de Hu para ambos objetos (implementación de la distancia Manhattan, ecuación 3), visible en la ecuación 7.

$$I_2(A, B) = \sum_{i=1...7} |m_i^A - m_i^B| \quad (7)$$

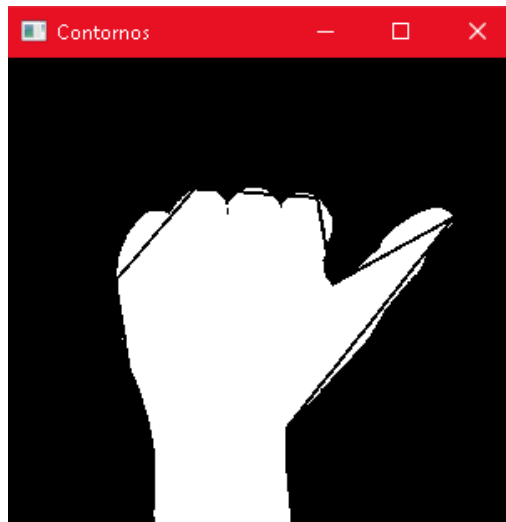
(OpenCV - Open Source Computer Vision, s.f.)

Por consiguiente, el mapeo realizado por el clasificador para este caso se considera concerniente al contorno de la mano signante. Dentro de las consideraciones para la comparación de las imágenes se tomó en cuenta, con el fin de obtener un porcentaje de coincidencia más útil, el establecer puntos específicos del contorno de cada una de las señas, derivando en resultados óptimos y valores de comparación menos aproximados (distancia natural), es decir, su cercanía con cero es más amplia. Se usó la delimitación de contornos, a través del método *CHAIN\_APPROX\_SIMPLE*, que, en comparación con los demás tipos de funciones, hace inferencia sobre puntos más apegados a la figura.

Estos puntos establecidos para las señas, a partir de la imagen ya binarizada, son ilustrados en la figura 3.20 para letra "A", como ejemplo.

### Figura 3.20

*Inferencia sobre puntos apegados a la figura de la seña*



Así mismo, la tabla 3.10 ilustra las funciones utilizadas en esta etapa y la figura 3.21 muestra el código de implementación de estas.

**Tabla 3.10**

*Funciones utilizadas en Python: Clasificación de imágenes*

Nombre	Descripción
<b>matchShapes(imagen_1, imagen_2, método, parámetro)</b>	Encuentra la distancia entre dos figuras a través de la diferencia entre los momentos de Hu de dos objetos o figuras.
<b>findContours(imagen, tipo_contorno, método)</b>	Define a detalle el contorno en una imagen binaria con el fin de facilitar el reconocimiento de la figura.

**Figura 3.21**

*Código de la obtención de la distancia entre momentos de Hu*

```
def distancia(imagen, imagen2):
#Imagen para comparar
    im1 = cv.imread("Images/PreImages/"+ imagen2, cv.IMREAD_UNCHANGED)
    ret, thresh = cv.threshold(im1, 127, 255, 0)
    ret, thresh2 = cv.threshold(imagen, 127, 255, 0)
    contours, hierarchy = cv.findContours(
        thresh, cv.RETR_TREE, cv.CHAIN_APPROX_SIMPLE)
    cnt1 = contours[0]
    contours2, hierarchy = cv.findContours(
        thresh2, cv.RETR_TREE, cv.CHAIN_APPROX_SIMPLE)
    cnt2 = contours2[0]
    m1 = cv.matchShapes(cnt1, cnt2, cv.CONTOURS_MATCH_I2, 0)
    print("Distancia entre " + imagen + " y " + imagen2)
    print((m1),end='\n')
imagen = cv.imread("Images/PreImages/A_1.jpg", cv.IMREAD_UNCHANGED)
```

Para la obtención de resultados a partir de las características descriptivas de las imágenes de la LSM se efectuaron pruebas a partir de los *frames* obtenidos de la cámara, contra las imágenes preprocesadas que ya se contenían en la BC (almacenada de forma local) para poder analizar la similitud respecto a las señas, tomando como

base lo postulado por *matchShapes*: entre menor sea la distancia resultante, las figuras en comparación tendrán más niveles de proximidad.

De aquí, que en la tabla 3.11 se muestran los resultados de la comparación de distancias entre cada una de las letras, para la obtención de un umbral de comparación.

**Tabla 3.11**

*Comparación de distancias de los momentos de Hu de las letras pertenecientes al abecedario de la LSM (A-C)*

Letra	A_1	A_2	A_3	A_4	A_5	B_1	B_2	B_3	B_4	B_5	C_1	C_2	C_3	C_4	C_5
A_1	0	0.474	0.395	0.383	0.84	2.701	3.199	3.158	2.304	1.934	1.909	1.551	1.505	1.936	2.227
A_2	0.474	0	0.325	0.841	0.714	2.227	2.725	2.839	1.83	1.674	1.435	1.232	1.186	1.624	2.063
A_3	0.395	0.325	0	0.613	0.446	2.552	3.05	3.152	2.155	1.974	1.76	1.545	1.499	1.925	2.364
A_4	0.383	0.841	0.613	0	1.059	3.068	3.566	3.362	2.671	11.97	2.276	1.755	1.709	12.12	11.95
A_5	0.84	0.714	0.446	1.059	0	2.451	2.949	3.194	2.053	2.16	1.659	1.587	1.541	2.111	2.55
B_1	2.701	2.227	2.552	3.068	2.451	0	0.498	0.975	0.397	1.015	1.154	1.135	0.756	0.852	1.295
B_2	3.199	2.725	3.05	3.566	2.949	0.498	0	0.887	0.895	1.37	1.319	1.222	0.95	1.314	1.651
B_3	3.158	2.839	3.152	3.362	3.194	0.975	0.887	0	1.247	1.522	1.98	2.109	1.731	1.385	1.497
B_4	2.304	1.83	2.155	2.671	2.053	0.397	0.895	1.247	0	0.864	1.052	1.106	0.543	0.585	1.144
B_5	1.934	1.674	1.974	11.97	2.16	1.015	1.37	1.522	0.864	0	1.266	0.587	0.401	0.463	0.438
C_1	1.909	1.435	1.76	2.276	1.659	1.154	1.319	1.98	1.052	1.266	0	0.335	0.26	1.208	1.647
C_2	1.551	1.232	1.545	1.755	1.587	1.135	1.222	2.109	1.106	0.587	0.335	0	0.563	0.841	0.612
C_3	1.505	1.186	1.499	1.709	1.541	0.756	0.95	1.731	0.543	0.401	0.26	0.563	0	0.357	0.466
C_4	1.936	1.624	1.925	12.12	2.111	0.852	1.314	1.385	0.585	0.463	1.208	0.841	0.357	0	0.695
C_5	2.227	2.063	2.364	11.95	2.55	1.295	1.651	1.497	1.144	0.438	1.647	0.612	0.466	0.695	0

Nota. Fragmento de la tabla de distancias entre cada una de las letras pertenecientes a la BC, las figuras en el *Ápndice B* denotan la tabla completa.

A partir de las distancias obtenidas para cada una de las letras, se definieron los valores del umbral, que permitieron realizar el análisis de las señas obtenidas respecto a las imágenes en la BC.

Por tanto, para la clasificación de las imágenes de acuerdo con los resultados obtenidos, las señas con configuraciones similares llegan a tener distancias demasiado cercanas, por ejemplo, para las letras “E” y “S”, ilustradas en las figuras 3.22 y 3.23.

Sin embargo, tras la realización de pruebas con el sistema el umbral aproximado para el reconocimiento de una seña con la función *matchShapes* ronda los valores de 0.7 a 0.9, por lo tanto, se decidió establecer un umbral igual o menor a 0.7 para determinar el reconocimiento de una seña como acertada, coincidiendo además con el promedio total de las medias para cada una de las 21 letras, que puede observarse en la tabla 3.12.

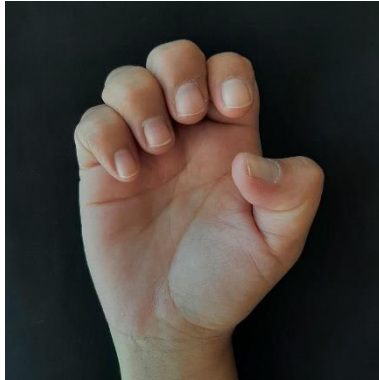
Es de esta forma que, al obtenerse los valores correspondientes a la comparación de distancia de la imagen obtenida respecto a las contenidas en la BC, que al encontrarse



por debajo del umbral o ser en su caso, igual al valor de éste se indica entonces que la imagen analizada es coincidente con la seña esperada, por lo cual es posible afirmar que el reconocimiento de la seña se considera exitoso.

**Figura 3.22**

*Letra "E"*



**Figura 3.23**

*Letra "S"*



**Tabla 3.12**

*Media de las distancias para cada letra del alfabeto estático de la LSM y media general de estas*

LETRA	MEDIA
A	0.40603566
B	0.64472587
C	0.46561698
D	1.60968797
E	0.52710031
F	0.49740176
G	0.85766676
H	2.47478236
I	0.58245252
L	1.21888897
M	0.74554224
N	0.51537337
O	0.87335037
P	0.92118869
R	0.67176777
S	0.45064102
T	0.302126
U	0.33700413

<b>V</b>	0.94343091
<b>W</b>	0.44336363
<b>Y</b>	0.68853338
<b>MEDIA GENERAL:</b>	0.77031813

### 3.3.5.3 Validación del Modelo.

Para poder verificar los resultados obtenidos con el uso de la función *matchShapes*, se llevó a cabo la comprobación de la distancia para la letra "A", desglosándose en las tablas 3.13 a 3.16.

La tabla 3.13 contiene la comparación de los 7 momentos de Hu de 2 de las 5 señas para la letra "A", almacenadas en la BC, con la finalidad de poder probar la veracidad de la distancia calculada mediante la función *matchShapes* versus el cálculo de la distancia natural realizada de forma manual.

**Tabla 3.13**

*Momentos de Hu para las letras "A" #1 y #2*

Letra: A	H0	H1	H2	H3	H4	H5	H6	Distancia (manual)	Distancia (MatchShapes)
<b>Original #1</b>	0.000792	8.25E-08	1.68E-10	6.96E-12	-2.32E-22	-1.39E-15	5.23E-23	0.013695396	0.013382
<b>Original #2</b>	0.000817	1.18E-07	1.20E-10	4.86E-12	-1.10E-22	-4.54E-16	4.25E-23		

Esta comprobación se realiza a partir de las ecuaciones 5, 6 y 7 mencionadas con anterioridad, dando como resultado la distancia natural entre las imágenes, en la tabla 4.14 podemos observar los signos designados por el vector correspondiente a los 7 momentos de Hu.

**Tabla 3.14**

*Signos correspondientes a los momentos de Hu*

$sign(h_i^A)$	1	1	1	1	-1	-1	1
$sign(h_i^B)$	1	1	1	1	-1	-1	1

Una vez determinados los signos, en lo que respecta a la tabla 3.15, se puede apreciar la operación logarítmica aplicada para cada uno de los 7 momentos de Hu.

**Tabla 3.15***Operación de logaritmo aplicada a los momentos de Hu*

$\log(h_i^A)$	-3.10112	-7.08365	-9.77568	-11.1577	-21.6351	-14.8578	-22.2818
$\log(h_i^B)$	-3.08774	-6.9271	-9.9202	-11.313	-21.96	-15.3431	-22.3714

Finalmente, en la tabla 3.16, se denota el cálculo de la diferencia entre los momentos de Hu de la seña de la letra “A” almacenada en la BC considerada como #1, respecto a la considerada como #2.

**Tabla 3.16***Diferencia entre momentos de Hu (“A” #1 con respecto a “A” #2)*

$m_i^A - m_i^B$	-0.01339	-0.15655	0.144518	0.15532	0.32488	0.485263	0.089621
$\Sigma$	1.36953960340146 x 10 <sup>-2</sup>						

Asimismo, dados los resultados el modelo se determina válido ya que, como se observó, el resultado de la distancia respecto a la función *matchShapes* es exactamente la misma, que, al realizarse el cálculo de forma manual, con una diferencia mínima en los decimales, con la ventaja de un cálculo más rápido con un menor uso de recursos.

### 3.4 Interfaz Gráfica

#### 3.4.1 Implementación en Computadora

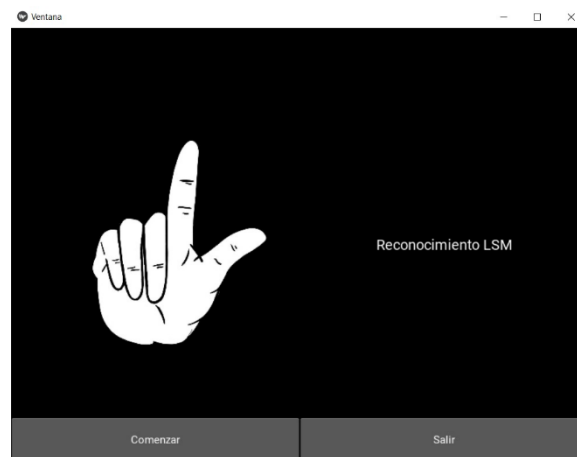
Con base a las características señaladas en la tabla 3.1, el desarrollo del aplicativo que reconoce las señas estáticas del abecedario de la LSM fue probado en primera instancia en este dispositivo.

##### 3.4.1.1 Menú Principal.

Se diseñó una interfaz gráfica de fácil acceso y uso, tomando en consideración la experiencia del usuario al utilizar términos prácticos y familiares, así como vistas simples e intuitivas. En el menú principal se situaron dos botones, uno que permite comenzar con el procesamiento de las señas y otro que finaliza aplicativo, justo como se muestra en la figura 3.24.

### Figura 3.24

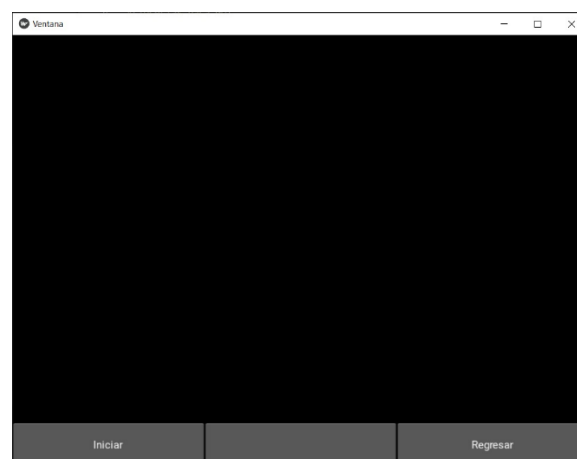
*Menú principal*



Consecutivamente en la figura 3.25 se visualiza la pantalla que despliega el botón “Comenzar”, la cual puede disparar dos tipos de acciones, la primera activa la cámara para iniciar con el reconocimiento de las señas estáticas del abecedario de la LSM, mientras que el segundo botón retorna al menú principal.

### Figura 3.25

*Pantalla de inicio: activa la cámara e inicia con reconocimiento de las señas*

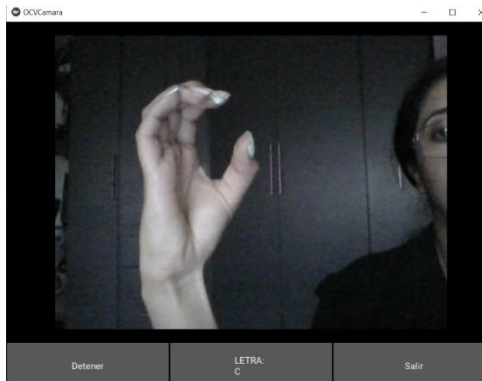


Una vez activado el botón “Iniciar”, automáticamente el aplicativo activa la cámara de la computadora para comenzar a comparar la seña captada en *tiempo real* contra los frames de las señas almacenados en la BC, de tal forma que, si la seña del signante corresponde a alguna de estas señas, la interfaz muestra la seña identificada, tal como se aprecia en el recuadro de texto identificado como “LETRA” en las figuras 3.26 y 3.27.

**Figura 3.26**

*Identificación de la letra “C”*

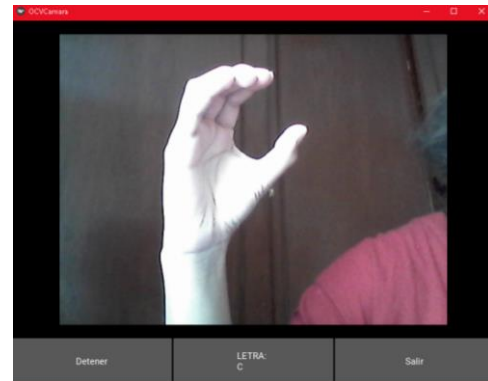
*en ambientes con poca luminosidad*



**Figura 3.27**

*Identificación de la letra “C”*

*en ambientes con bastante luminosidad*



### **3.4.2 Implementación en Dispositivo Móvil**

El diseño y la implementación del aplicativo destinado a ser utilitario en dispositivos móviles con sistema operativo Android, es totalmente amigable con Kivy ya que permite el desarrollo con la misma estructura de Python evitando la dependencia de emplear un emulador para su funcionamiento.

Para la implementación del aplicativo dentro del dispositivo móvil, se realizó una transferencia bluetooth, siendo este un archivo ejecutable de extensión .apk que permite instalar aplicaciones para Android. Este ejecutable requiere la aprobación del usuario tal como se muestra en la figura 3.28 y posterior a ello el proceso de instalación en la figura 3.29.

**Figura 3.28**

*Aprobación de instalación*



**Figura 3.29**

*Instalación de la aplicación*



### **3.4.2.1 Menú principal**

Una vez se encuentra instalada la aplicación, se puede observar la pantalla de inicio, que da la bienvenida al usuario y permite detonar las acciones de procesamiento de las señas en tiempo real, los detalles gráficos pueden observarse en la figura 3.30 y en la figura 3.31 se resalta el ícono que permite identificar al aplicativo.

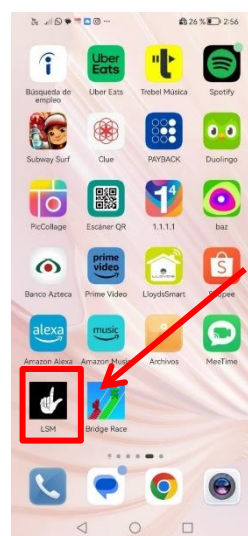
**Figura 3.30**

*Menú principal*



**Figura 3.31**

*ícono de app LSM*



En relación con las figuras precedentes, se puede resaltar que la aplicación adopta dos funciones básicas, o comienza con la interpretación y reconocimiento de señas de la LSM o se sale permitiendo visualizar el ícono del aplicativo.

Al presionar la opción “Comenzar”, el sistema solicita los permisos necesarios para acceder al dispositivo de entrada del móvil (cámara y contenido multimedia – almacenamientos externo e interno) tal como se aprecia en la figura 3.32 y 3.33.

**Figura 3.32**

*Acceso a contenido multimedia*



**Figura 3.33**

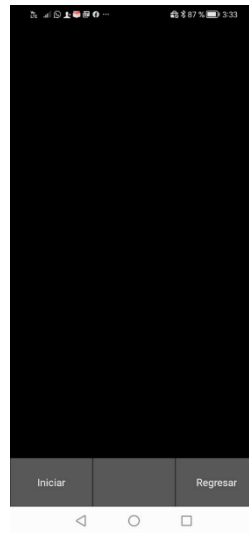
*Acceso para grabar y tomar fotos*



Análogamente, a la aprobación de permisos, se despliega una interfaz que permite disparar dos acciones, la primera identificada con el botón “Iniciar”, permite realizar la captura de la seña en *tiempo real* y proceder al reconocimiento de la seña haciendo una comparativa de la seña con nuestra BC. Y el botón “Regresar” que despliega el menú principal del aplicativo. Lo antes mencionado puede visualizarse en la figura 3.34.

### Figura 3.34

*Pantalla de inicio: activa la cámara e inicia con reconocimiento de las señas*





## CAPÍTULO 4. PRESENTACIÓN Y DISCUSIÓN DE RESULTADOS

Este capítulo resume los resultados obtenidos tras ejecutar las pruebas pertinentes con el sistema desarrollado, así como el análisis de estos para determinar el cumplimiento de la hipótesis.

### 4.1 Resultados del Reconocimiento del Alfabeto Estático de la LSM

El usar principalmente los datos recabados en la tabla 3.9, fue la base para llegar a la función *matchShapes*, cuyo propósito es minimizar el mecanismo de comparación de datos para cada seña, al realizarlo en menor tiempo y de manera eficaz.

En cuanto al desempeño del sistema, se realizaron 100 pruebas documentadas para determinar la eficiencia del reconocimiento de patrones de la LSM, estas se llevaron a cabo en dos tipos de ambientes: con abundante y poca luminosidad. Por lo cual, se efectuaron 50 pruebas por cada uno de los ambientes, dando el total de 100 para cada seña estática del abecedario de la LSM, resultando en un total de 2100 pruebas.

Por ejemplo, en la figura 4.1 se desglosa el análisis de las pruebas realizadas en presencia de bastante luminosidad definida como *luz*, representando los valores referidos como 1 = Sí, es decir, la seña fue reconocida correctamente y 0 = No, es decir, la seña no pudo reconocerse satisfactoriamente.

Así mismo, la figura 4.2 denota el análisis de las pruebas llevadas a cabo en presencia de poca luminosidad, determinándose como *oscuridad*, bajo la misma premisa de valores 1 = Sí y 0 = No.



Es así como, a partir del procedimiento de pruebas, en donde para determinar que una imagen coincide de manera precisa con la seña predeterminada; es factible decir que todas las señas fueron reconocidas difiriendo únicamente en el porcentaje de coincidencia de cada una de éstas respecto a la exposición a la luz del ambiente. En la tabla 4.1 es visible el porcentaje de coincidencia para cada letra de las señas estáticas en la LSM.

Los porcentajes obtenidos para cada ambiente se definen de acuerdo con la coincidencia de las señas obtenidas con la cámara en *tiempo real* en comparación con las imágenes de la BC, tomando como 100% la identificación de todas las señas, y siendo el total, la suma de los porcentajes obtenidos en el ambiente correspondiente.

Con relación a los resultados obtenidos, es factible afirmar que las señas correspondientes a las letras “A” y “W” de la LSM, presentan los porcentajes de coincidencia más altos, con 92% y 98% respectivamente, en la tabla 4.1 son concebibles los datos porcentuales para cada una de las letras de la LSM consideradas en este trabajo de estudio.

En caso contrario, en lo que concierne a la seña con menor porcentaje de éxito se encuentra la letra “M”, con un 36% de coincidencia, debido a la similitud que presenta con su subsecuente, la letra “N”, que presentó mejores alcances de identificación.

Por lo que se refiere al análisis de los datos obtenidos a partir de las figuras 4.4 y 4.6, es notorio que, la mayoría de las señas que componen el estudio efectuado obtuvieron porcentajes de correcto reconocimiento superiores a 60%.

Por otra parte, el porcentaje con mayor ocurrencia en ambientes con bastante luminosidad es de 76% mientras que, en ambientes considerados oscuros, el porcentaje con mayor frecuencia es de 66%.

De manera general, el sistema reconoce acertadamente para cada ambiente, el 74% de las señas en ambientes de oscuridad, en contraste con el 73% en ambientes con luminosidad basta, obteniéndose un 74% total de reconocimiento del alfabeto estático de la LSM.

Con base en estos porcentajes, se destaca el empleo del sistema en ambientes oscuros a través de la preeminente identificación del mayor porcentaje de las señas del alfabeto estático de la LSM, permitiendo la mejor distinción de la mano derecha, puesto que se evitan intercesores en el fondo (brillo, reflejo, sombra).

Sin embargo, cabe resaltar que los porcentajes para cada ambiente son muy cercanos entre sí, apuntando a que el sistema tiene la capacidad de fungir en cualquiera de estos (ambos considerados por las condiciones de día y noche).

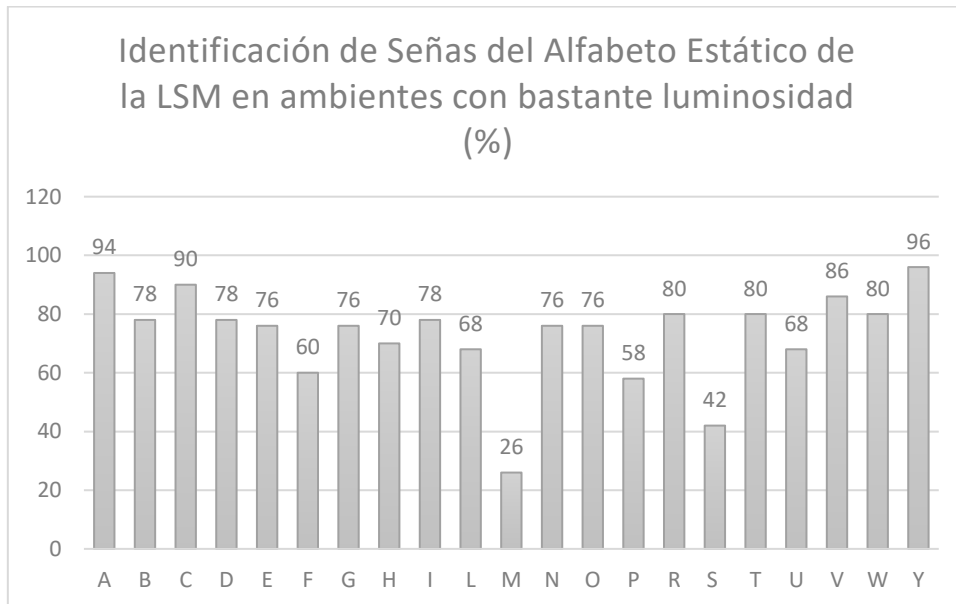
**Tabla 4.1**

*Porcentaje de coincidencia en el reconocimiento de las señas estáticas correspondientes a la dactilología de la LSM*

<b>Letra</b>	<b>Oscuridad (%)</b>	<b>Luz (%)</b>	<b>Total (%)</b>
<b>A</b>	90	94	92
<b>B</b>	100	78	89
<b>C</b>	88	90	89
<b>D</b>	66	78	72
<b>E</b>	78	76	77
<b>F</b>	62	60	61
<b>G</b>	66	76	71
<b>H</b>	72	70	71
<b>I</b>	44	78	61
<b>L</b>	66	68	67
<b>M</b>	46	26	36
<b>N</b>	70	76	73
<b>O</b>	88	76	82
<b>P</b>	56	58	57
<b>R</b>	72	80	76
<b>S</b>	66	42	54
<b>T</b>	80	80	80
<b>U</b>	86	68	77
<b>V</b>	80	86	83
<b>W</b>	80	80	80
<b>Y</b>	100	96	98
<b>Total</b>	74	73	74

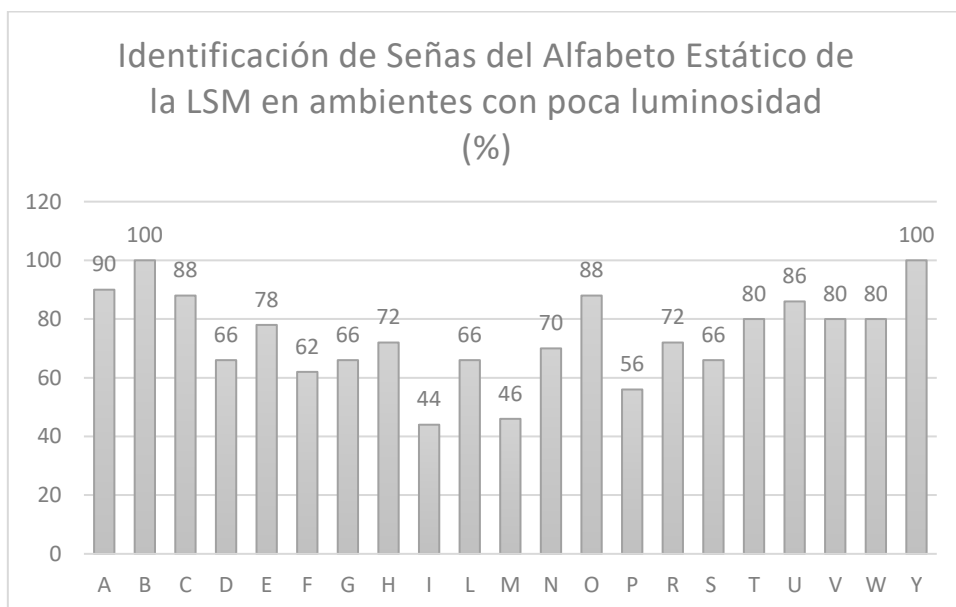
**Figura 4.3**

*Gráfica de Identificación de Señas del Alfabeto Estático de la LSM en ambientes con bastante luminosidad*



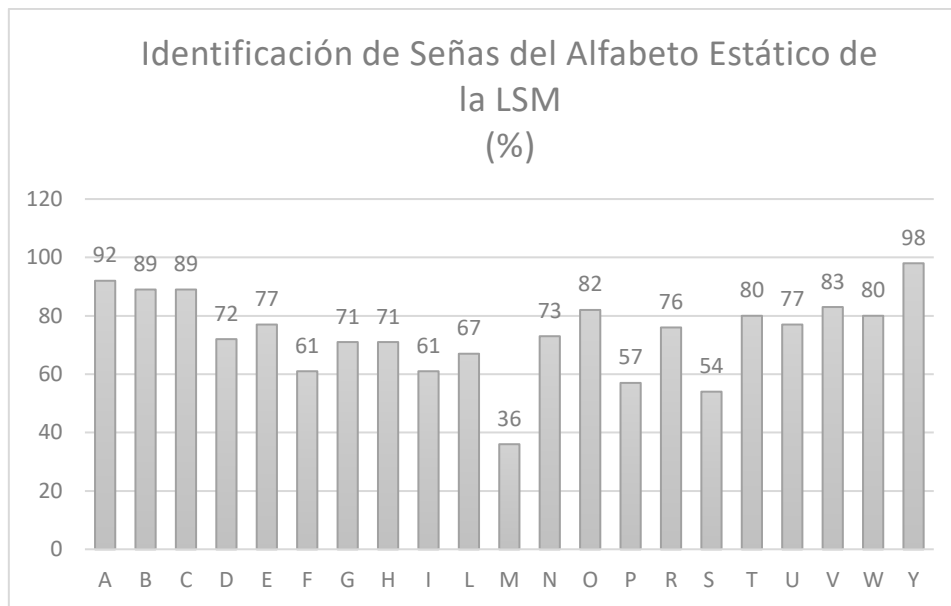
**Figura 4.4**

*Gráfica de Identificación de Señas del Alfabeto Estático de la LSM en ambientes con poca luminosidad*



**Figura 4.5**

*Gráfica de Identificación de Señas del Alfabeto Estático de la LSM*



#### **4.2 Análisis de Precisión**

Este análisis es realizado con el fin de depurar la definición de precisión del sistema en cuestión, a través de la Media Geométrica (MG), denotada en la ecuación 8, que se reconoce por ser una medida de posición central que permite la observación de un conjunto de datos para su estudio.

$$MG = \sqrt[n]{x_1 \cdot x_2 \cdot \dots \cdot x_n} \quad (8)$$

(Economipedia, 2020)

Donde:

n: Es el número total de observaciones realizadas;

x: Variable sobre la que se calcula la MG;

Adicionalmente, puede agregarse una posición a cada observación (N), determinada generalmente como *i*.

Para este caso particular, la MG se desglosa de la forma en que se visualiza en la ecuación 9, y el resultado de esta se percibe en la ecuación 10.

$$MG = \sqrt[21]{92 \cdot 89 \cdot 89 \cdot 72 \cdot 77 \cdot 61 \cdot 71 \cdot 71 \cdot 61 \cdot 67 \cdot 36 \cdot 73 \cdot 82 \cdot 57 \cdot 76 \cdot 54 \cdot 80 \cdot 77 \cdot 83 \cdot 80 \cdot 98} \quad (9)$$

Es observable que cada  $x$  denota el porcentaje de coincidencia obtenido para cada letra, y  $n$  es el número de elementos, en este caso las señas analizadas.

$$MG = 72.06 \quad (10)$$

Dado este resultado, se tiene una **precisión del 72.06% en la identificación acertada** de las letras pertenecientes al alfabeto estático de la LSM. Se señala que en la etapa de aprendizaje se obtuvo una precisión del 52.8%.

### 4.3 Limitaciones Encontradas

A continuación, se enlista una serie de limitaciones que afectaron al reconocimiento del LSM.

#### 4.3.1 Fondo e Iluminación

Una de las principales limitaciones del sistema se denota en el espacio o ambiente donde son recolectadas las imágenes. Se presentó en un principio la captura de imágenes en un ambiente "natural", con fondo en colores claros; sin embargo, el procesamiento para estos primeros casos resultó en imágenes con altos niveles de ruido (consecuencia de sombras o reflejos de la luz sobre el fondo elegido). Por lo tanto, se definió el uso de fondos en colores oscuros u opacos donde los resultados al preprocesamiento digital fueron satisfactorios en relación con lo esperado para el trabajo con las imágenes.

Tras la realización de las primeras pruebas del sistema respecto al uso de los Momentos de Hu y obtención de distancias, una de las características que se resaltaron fue el nivel de iluminación en el espacio definido.

Como se ha mencionado, el reflejo en el fondo de las imágenes así como el que haya figuras adicionales en este mismo implica siluetas añadidas dentro del espacio establecido para el tratamiento de éstas, por lo que se determinó la ejecución de pruebas en ambientes con distintos tipos de iluminación e incluso la ausencia de luz natural, obteniéndose resultados sobresalientes en entornos con poca luminosidad o luz natural demasiado tenue, lo que permite que la seña sea determinada de una manera óptima.

En la figura 4.6, pueden observarse las diferencias de binarización de imágenes en presencia de luz natural (brillo alto) versus el ambiente con poca presencia de luz natural.

**Figura 4.6**

*Pruebas con presencia de luz natural*



La seña no se obtiene directamente debido a la interferencia de la luz reflejada en el fondo.

Es notable el cambio en la figura 4.7, donde la definición de la silueta es notoria al encontrarse en distintos tipos de iluminación.

**Figura 4.7**

*Pruebas con ausencia de luz natural*



Dependiendo de la cantidad o intensidad de luz, se obtienen distintos resultados de las imágenes adquiridas y su reconocimiento en comparación con la seña. Dichos resultados son apreciables en la figura 4.8.



**Figura 4.8**

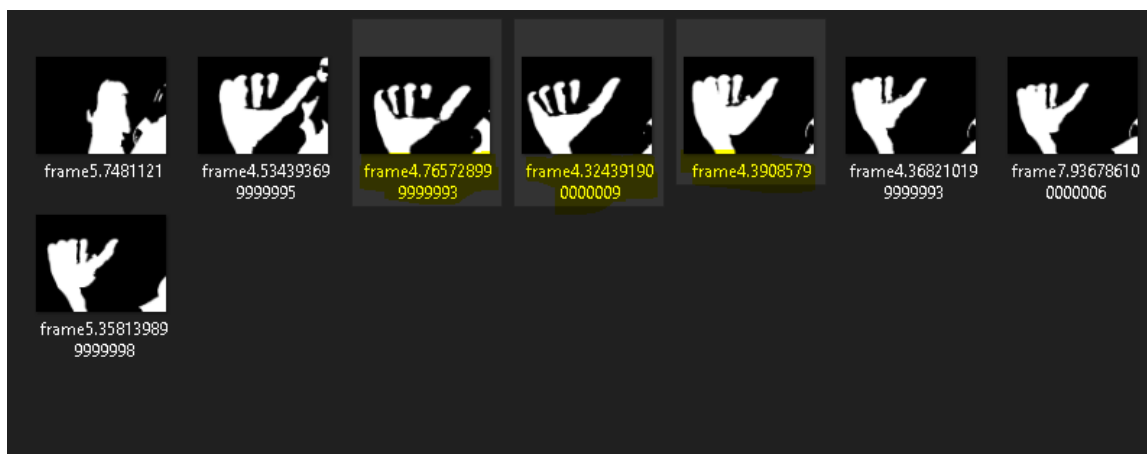
*Pruebas con distinta intensidad de luz natural*



Como se puede apreciar en la figura 4.9, en mayor presencia de luz y dependiendo del ángulo de ésta, la seña va perdiendo las características que la denotan. Aún en estas condiciones, el sistema fue capaz de reconocer 6/7 señas, validándose la letra correspondiente, en este caso, la “A”.

**Figura 4.9**

*Frames resultantes con distinta intensidad de luz*



### **4.3.2 Dispositivo para la Captura de Imágenes**

Debido a que el principal objetivo de la presente investigación es dirigido a la obtención de imágenes, el dispositivo a cargo de esta tarea representa un cierto nivel de limitación. Es sabido que entre mayor calidad se obtenga de un aparato para la captura de imágenes, mejor serán los detalles representativos de estos archivos, en consecuencia,

es importante el mencionar, que mientras mejor se encuentren las propiedades del dispositivo a ser utilizado para la ejecución del sistema, mejores serán los resultados obtenidos y, consecuentemente, darán más precisión.

### **4.3.3 Uso del Descriptor Elegido**

Como su definición lo indica, los momentos de Hu se encargan de obtener valores característicos de la imagen, independientemente de su tamaño y orientación, por lo que, al ser algunas señas representadas con configuraciones similares, el resultado obtenido al leerse en *tiempo real* puede coincidir con otras letras. Este efecto causaría que el porcentaje de coincidencia para algunas letras disminuya al presentarse en el sistema el caso de señas con valores de distancias similares o muy cercanas.

De acuerdo con los resultados obtenidos en las 100 pruebas de la detección de señas del alfabeto estático de la LSM, pudimos observar que para la letra “M”, cuyo porcentaje de coincidencia fue el más bajo; uno de los factores influyentes sobre este resultado fueron las complicaciones por la iluminación, añadido a que la configuración es similar con respecto a la letra “N” tal como se muestra en la figura 4.10.

Otra de las señas que presentan estos atributos de configuración similar son las correspondientes a la letra “S” con respecto a la letra “E”, la letra “A” y en algunos casos la letra “I”, esto se ilustra en la figura 4.11.

Finalmente, en la figura 4.12, se puede percibir que la letra “U” y la letra “V”, también son consideradas con características de signado similar.

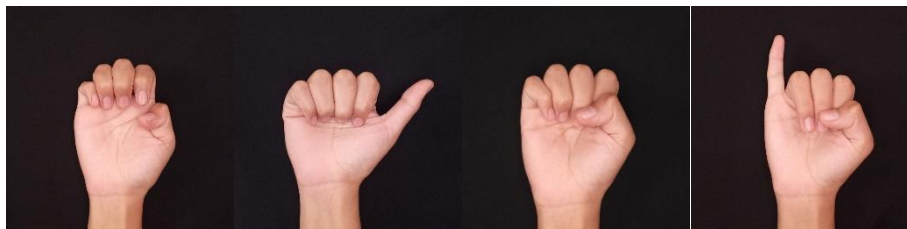
#### **Figura 4.10**

*Letra “M”, respecto a la letra “N” (configuración similar)*



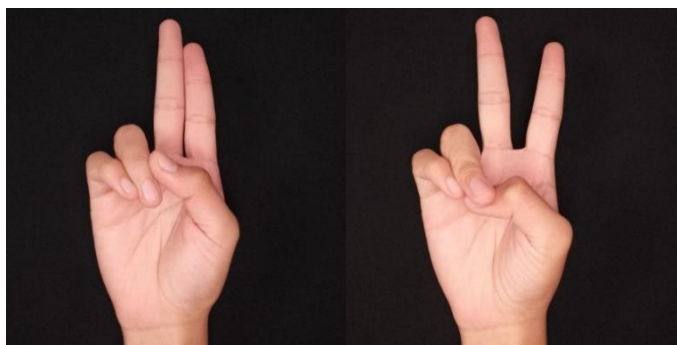
#### Figura 4.11

Letra "S", respecto a la letra "E", letra "A" y en algunos casos a la letra "I"  
(configuración similar)



#### Figura 4.12

Letra "U", respecto a la letra "V" (configuración similar)

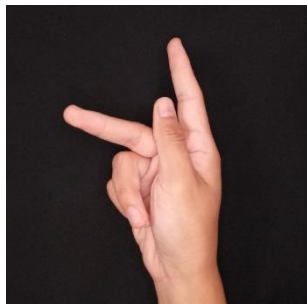


#### 4.3.4 Configuración de las Señas

Considerado como uno de los retos con mayor relevancia al tratarse ambos miembros que presentan esta investigación el no ser personas signantes, el poder realizar cada una de las señas según lo establecido en el Diccionario de Lengua de Señas Mexicana y ser reconocida por el sistema en el tiempo establecido, fue una más de las dificultades encontradas, puesto que, las manos predisponían de cierta rigidez al tratar de acostumbrarse a la disposición de cada una de las señas, principalmente aquellas que implican el mantenerse en una posición de más de tres puntos de flexibilidad diferentes. Ejemplo son las configuraciones de la seña para las letras: "C", "F", "P", "M" y "W". Se ilustra en la figura 4.13 la configuración para la letra P.

### Figura 4.13

*Seña de la letra P*



### 4.4 Comparación de los Resultados

De acuerdo con sistemas previamente realizados para el reconocimiento de señas, se resalta principalmente el uso de un dispositivo móvil para efectuar la ejecución del sistema para el presente tema de investigación, obteniendo los tiempos descritos en la tabla 4.3 para cada acción.

**Tabla 4.3**

*Tiempos de ejecución del sistema*

<b>Acción</b>	<b>Tiempo</b>
<b>Inicio de cámara</b>	2 segundos
<b>Reconocimiento de seña</b>	15 a 20 milisegundos
<b>Detenimiento de aplicación/eliminado de frames</b>	3 segundos
<b>Cierre de aplicación</b>	1 segundo

Además, la distancia promedio entre la mano del señante y la cámara debe ser entre 30 y 50 centímetros dependiendo de la configuración de la seña para la mejor captura de ésta.

Asímismo, en la tabla 4.4 son denotables las diferencias entre trabajos similares previos, que emplean tecnologías de entrenamiento y auto aprendizaje en comparación con el sistema desarrollado en este trabajo de investigación.

**Tabla 4.4**

*Comparación de resultados obtenidos respecto a trabajos previos*

<b>Responsables del trabajo (Por fecha)</b>	<b>Descripción del trabajo</b>	<b>Patrones reconocidos</b>	<b>Precisión (%)</b>	<b>Clasificación</b>	<b>Delimitaciones/Observaciones</b>
Cuecuecha et al. (2015)	Reconocimiento de vocales signadas (siendo todas ellas estáticas), por medio del uso de <i>Leap Motion</i> a través de dispositivos con sensores de reconocimiento de gestos, como <i>Kinect</i> . El número total de muestras fue de 100 registros.	5 letras	100%	<i>Deep Learning</i>	Ninguno hay entrecruce alguno o similitud entre las configuraciones, sin embargo, el porcentaje de precisión disminuye a medida que se aumenta el número de letras a ser identificadas.
Hernández de la Luz et al. (2018)	Con uso de la tecnología <i>Leap Motion</i> y tecnología 3D se creó un prototipo, para impulsar una base para el desarrollo de un intérprete de LSM. El número total de muestras fue de 2002 registros.	16 letras	95%	KNN	En las letras R, U, V, F, O. Debido a la posición de los dedos, entrecruzados o posicionados contra la palma de la mano.
Hernández (2019)	Uso de <i>Kinect</i> y <i>Leap motion</i> para la captura de gestos para la posición y desplazamiento de los brazos.	32 gestos	95%	KNN	Mínimo de errores, debido al color de piel de las personas e iluminación en los ambientes determinados para las pruebas, así como ciertas

	El número total de muestras fue de 4672 registros.					posiciones que representaron obstáculos de identificación, como el entrecruce de dedos o entrelaces.
Trujillo y García (2021)	Uso de sensores RGB, por medio de redes neuronales perceptrón multicapa y algoritmo BackPropagation. El número total de muestras fue de 1590 registros.	53 palabras	90%-99%	K-fold cross validation		Mínimo de errores, debido a la consideración de personas signantes con distintos tonos de piel
Presente trabajo (2023)	Uso de cámara de dispositivos: laptop y teléfono. El número total de muestras fue de 105 registros.	21 letras	72%	KNN		Bajo porcentaje en las letras M, P, S. Debido a la similitud en la configuración de algunas letras y la iluminación del entorno en el que las pruebas fueron desarrolladas.

#### 4.5 Cálculo de Fiabilidad

El cálculo estadístico de fiabilidad se llevó a cabo con el software IBM SPSS, en la figura 4.14 se puede observar el análisis de fiabilidad.

Para este caso, para el reconocimiento promedio de la LSM hallado, tal como se muestra en la figura 4.15. Se observa que todos los constructos calculados en esta investigación **son confiables** ya que sus valores  $\alpha$  son mayores a 0.7, teniendo en valor de  $\alpha$  superior a 0.9 para el promedio del reconocimiento de las señas.

Figura 4.14

Cálculo de fiabilidad para detección en oscuridad y con iluminación

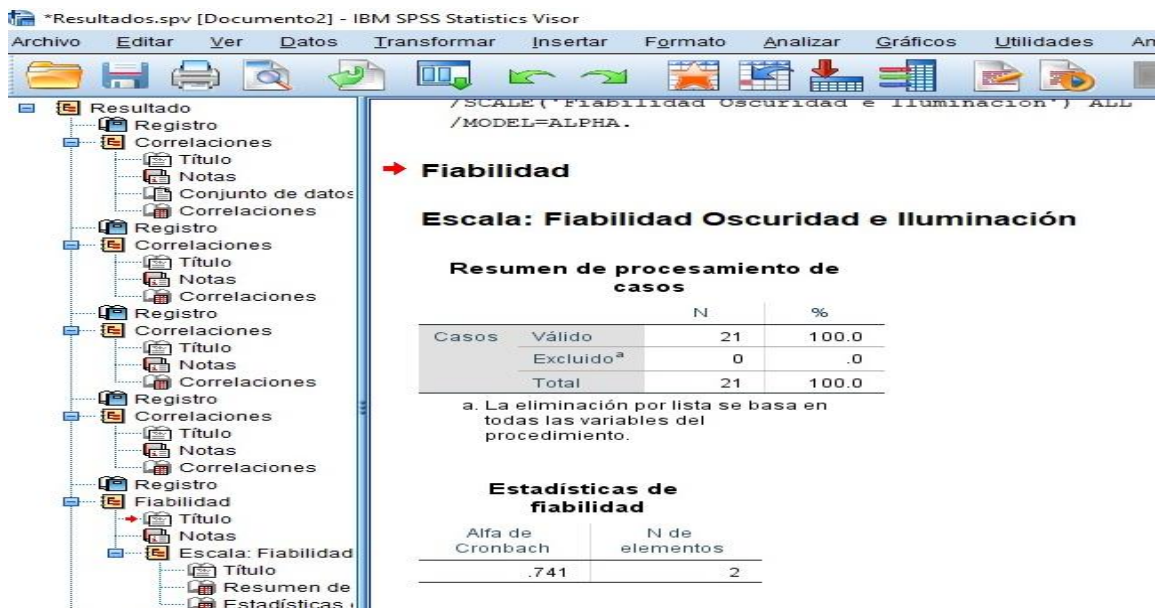
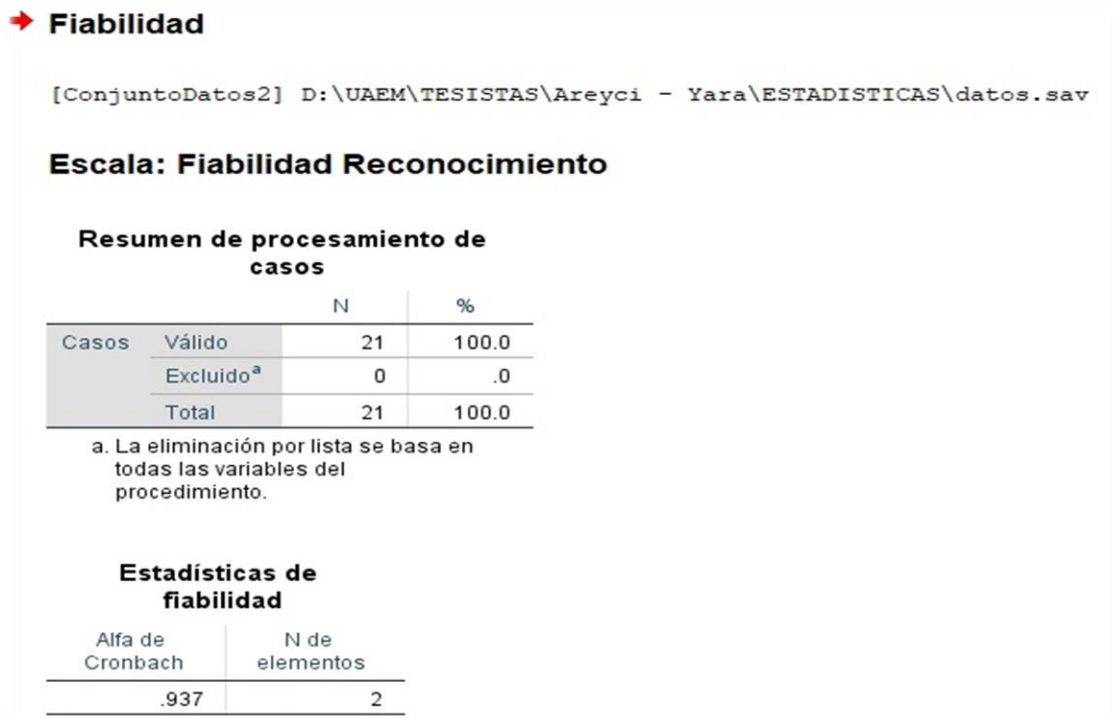


Figura 4.15

Cálculo de fiabilidad del valor promedio de reconocimiento del LSM hallado



## 4.6 Prueba de Hipótesis

Por otro lado, como puede observarse en la tabla 4.1, el porcentaje total de coincidencia de reconocimiento de las señas es de 74% en ambientes considerados oscuros y 73% en luz natural, alcanzando un 74% total, así como precisión del 72%. Lo que comprende y afirma que el sistema permite el reconocimiento de patrones con uso de los momentos invariantes de Hu, siempre y cuando el fondo permita la distinción de la configuración de la mano derecha, evitando en lo más posible la interferencia de ruido por reflejos u objetos ajenos al ambiente elegido para el reconocimiento.

La comprobación de hipótesis se realizó a través del coeficiente de correlación de Pearson o coeficiente de correlación lineal, cuyo análisis determina la relación entre dos variables.

De manera general, se realiza la división de la covarianza de las variables a ser estudiadas, y el producto de esta respecto a la raíz cuadrada del producto de sus varianzas. Es así como se intenta cuantificar la dependencia lineal entre dos variables aleatorias cualitativas.

El cálculo de correlación se efectúa por medio de la ecuación 11.

$$\rho_{XY} = \frac{Cov(X.Y)}{\sqrt{Var(X) \cdot Var(Y)}} \quad (11)$$

$$-1 \leq \rho_{XY} \leq 1$$

(Probabilidadyestadistica.net, s.f.)

La hipótesis general indica que *el reconocimiento de patrones del alfabeto estático de la LSM está positivamente asociado con el uso de momentos invariantes de Hu*. De aquí que se construyeron en IBM SPSS las variables VIO y VIL conteniendo los constructos referentes al reconocimiento de las señas en la oscuridad y en luz natural, mientras que la variable VD como variable dependiente conteniendo los valores promedio del reconocimiento.

Los resultados obtenidos por el programa IBM SPSS son los mostrados en la figura 4.19a y 4.19b, donde se observa una correlación significativa del 90.3% y del 91.4% y un valor de significancia 0.0.



**Figura 4.19a**

*Prueba de hipótesis considerando ambientes con poca iluminación*

### Correlaciones

		OSCURIDAD	RECONOCIMIENTO
OSCURIDAD	Correlación de Pearson	1	.903**
	Sig. (bilateral)		.000
	N	21	21
RECONOCIMIENTO	Correlación de Pearson	.903**	1
	Sig. (bilateral)	.000	
	N	21	21

\*\* . La correlación es significativa en el nivel 0,01 (bilateral).

**Figura 4.19b**

*Prueba de hipótesis considerando ambientes con bastante iluminación*

### → Correlaciones

		LUZ	RECONOCIMIENTO
LUZ	Correlación de Pearson	1	.914**
	Sig. (bilateral)		.000
	N	21	21
RECONOCIMIENTO	Correlación de Pearson	.914**	1
	Sig. (bilateral)	.000	
	N	21	21

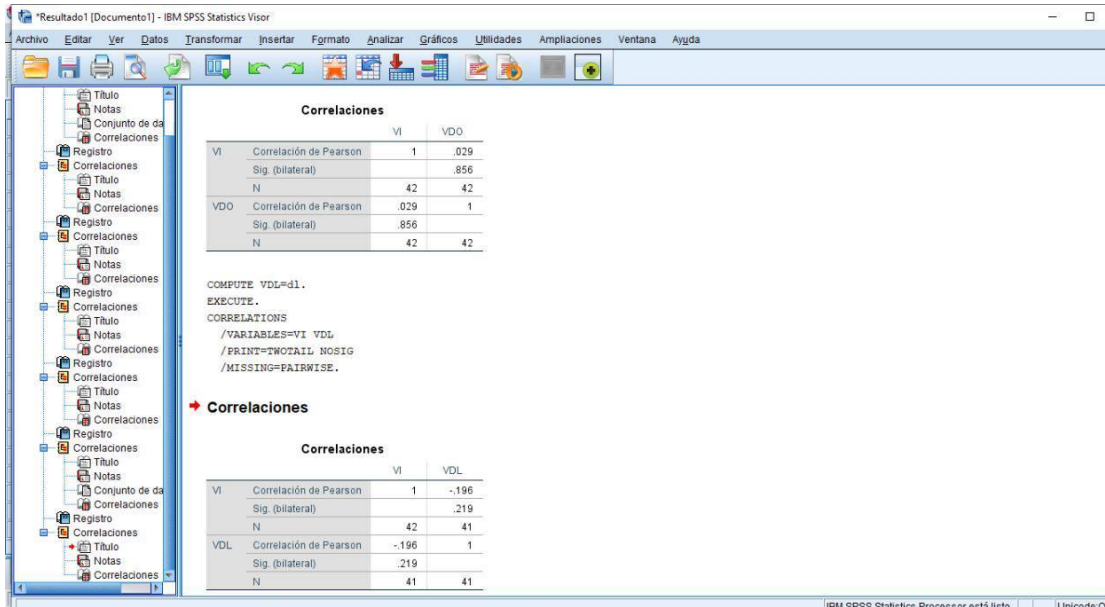
\*\* . La correlación es significativa en el nivel 0,01 (bilateral).

De igual manera, se elaboró un análisis de correlación de Pearson donde se distingue que **no hay correlación** de los momentos de Hu con respecto a las distancias obtenidas para el reconocimiento de cada una de las señas, es decir, no afectan o modifican las distancias para que no se pueda realizar el reconocimiento de las señas tal como se muestra en la figura 4.20.

Dados los resultados anteriores se concluye que los momentos invariantes de HU están positivamente asociados al reconocimiento del alfabeto estático de la LSM, por tanto, **la hipótesis general de investigación se acepta** y la hipótesis nula se rechaza

**Figura 4.20**

*Prueba de correlación entre momentos de Hu y distancia de adquisición de la seña*



Donde:

$V_I$  = Variables Independientes -> Momentos de Hu;

$V_{DO}$  = Variable dependiente -> Valores de distancias en Oscuridad;

$V_{DL}$  = Variable dependiente -> Valores de distancias con Luz

## CONCLUSIONES

Por medio del análisis de los resultados obtenidos a través de este trabajo de investigación, el uso de descriptores gráficos pudo ser comprobado como una herramienta significativa y justa de acuerdo con los requerimientos establecidos para el reconocimiento de patrones de la LSM al efectuarse la correspondencia de seña obtenida en *tiempo real* contra la letra correspondiente al abecedario estático de la LSM, con implementación de los momentos invariantes de Hu y con abordaje sobre dispositivos móviles.

La técnica utilizada para el análisis de los resultados obtenidos y de la investigación se determina de tipo cuantitativa – correlacional, puesto que se pretendió el relacionar el uso de métodos supervisados en un sistema tecnológico con el fin de justificar si su implementación es considerada adecuada para el reconocimiento de patrones en la Lengua de Señas Mexicana.

Por lo tanto, para la pregunta de investigación planteada *¿La aplicación de los momentos invariantes de Hu permiten el reconocimiento de patrones del alfabeto estático de la LSM con implementación en dispositivos móviles?*, y conforme a los porcentajes de identificación de las señas superiores al 70%, es razonable el determinar que **es posible su aprovechamiento para identificar las letras** correspondientes al abecedario de la LSM.

Sin embargo, se sugiere considerar el presente sistema como fundamento para futuros desarrollos de reconocimiento por medio de dispositivos móviles, con el fin de obtenerse resultados en menor tiempo y porcentajes de identificación de señas más altos y veraces; al ser éste susceptible a un mejor reconocimiento a medida de su mayor uso y de los medios de captura de la imagen.

Ahora bien, con base al desarrollo del sistema para el reconocimiento de patrones del alfabeto estático de la LSM con aplicación de los momentos invariantes de Hu, e intencionado a implementarse en dispositivos móviles con el fin de representar un uso con mayor alcance entre la comunidad sorda mexicana fue satisfactorio, logrando un 74% de certeza en la identificación de las señas estáticas en la dactilología de la LSM y 72% de precisión; además que el tiempo de reconocimiento establecido, no supera los veinte milisegundos, siendo este tiempo una importante ventaja al ser un sistema destinado a dispositivos móviles. Por tanto, **el objetivo planteado se cumple**.

Se destaca también un rendimiento eficaz en ambientes con poca luminosidad, debido a que este tipo de ambiente representa presencia de ruido en las imágenes mínimo o nulo, y de esta forma, la definición de la seña representada se obtiene con mayor nitidez. De aquí que los dos primeros objetivos particulares: *reconocer y capturar imágenes estáticas por medio de la cámara del dispositivo móvil con presencia de luz y, mejorar la nitidez y los elementos de la imagen y eliminar aquellos rasgos indeseables para un mejor procesamiento de esta*, se cumplen.

En referencia al tercer objetivo particular, *identificar la seña captada en la imagen con base en los patrones almacenados en la BC conocida como banco de imágenes* se cumple dados los porcentajes de similitud hallados al realizar la comparación de distancias entre las señas capturadas y las almacenadas en la BC.

Así mismo, el cuarto objetivo particular, *desarrollar una interfaz amigable para un fácil manejo de la aplicación móvil* se cumplió al desarrollar un sistema que se instala como cualquier otra aplicación, y cuya interfaz es intuitiva.

La *prueba de hipótesis* por correlación de Pearson que, considerando como variables dependientes al reconocimiento del LSM y como independientes a los momentos de Hu como descriptores geométricos, arrojó un resultado de correlación significativa superior al 90% con significancia del 0.0, por tanto la hipótesis general de investigación *La aplicación de descriptores gráficos está positivamente relacionada con el RP, en tiempo real, de la LSM bajo un enfoque supervisado se acepta y la hipótesis nula se rechaza*.

Así mismo, se puede afirmar que este trabajo representa una aproximación para la construcción de un traductor más completo de la LSM a implementarse en dispositivos móviles.

### **Aporte computacional**

En la actualidad, Python ha tomado gran relevancia en el mundo de la tecnología por las ventajas que representa al tratarse de un lenguaje dedicado ciertamente al procesamiento y análisis de datos; siendo en este caso el enfoque a la evaluación y clasificación de imágenes.

Su uso en este trabajo de investigación se considera como uno más de los medios para acercar a este lenguaje con el desarrollo móvil sin uso de simuladores para su testeo, permitiendo de esta forma la implementación de un mismo sistema en dos dispositivos de características diferentes y para ambientes distintos, considerando la portabilidad entre uno y otro, sin más que un intérprete o “traductor” de por medio para Windows-Android.

En la rama de RP puede precisarse el uso de estas herramientas puesto que, las características implementadas en este trabajo denotan la ventaja en cuanto a tiempos de ejecución y operacionales, pudiendo conformar un sistema básico con implementación de librerías que no consideran alguna red neuronal o entrenamiento previo.

### **Aporte social**

Como se predispuso en capítulos anteriores, la población sorda sigue encontrándose con la dificultad de desarrollarse y desenvolverse en el ambiente y sociedad como la conocemos, sin las herramientas adecuadas.

Por consiguiente, y después de querer aminorar el obstáculo que la comunicación representa a la comunidad sorda, el presente sistema es un escalón fundamental para poder reducir la brecha de diálogo al ser más probable en esta época el tener acceso a un dispositivo móvil y el adecuar un método de conversación por medio de estos, partiendo del alfabeto de la LSM, usado principalmente en la dactilología.

## SUGERENCIAS

A pesar de que los resultados obtenidos en el reconocimiento de la LSM con aplicación de los momentos invariantes de Hu han sido aceptables, esta metodología puede mejorarse a partir de las sugerencias siguientes, con el objetivo de incrementar la precisión del sistema:

- Utilizar imágenes de las manos sin fondo uniforme y con diferentes iluminaciones para establecer procedimientos adicionales en las etapas de procesamiento y segmentación.
- Experimentar con un número mayor de imágenes en la BC para poder incrementar la población muestra de la que depende la extracción de características para el posterior reconocimiento de imágenes capturadas en *tiempo real*.
- Utilizar en conjunto otro tipo de descriptores, con la finalidad de precisar las características de las imágenes y reducir el margen de reconocimiento incorrecto de las señas.
- Incorporar funcionalidades de IA como el empleo de redes neuronales, en conjunto con el aprendizaje automático, con el objetivo de permitir facilitar el proceso de RP.

## REFERENCIAS

- Andrés V., L. (2014). *Breve historia de las personas con discapacidad: de la opresión a la lucha por sus derechos*. Rebelión. <https://rebellion.org/docs/192745.pdf>
- Araiza, A. B., Díaz, D. A., & Segundo, L. M. (2019). Herramienta en realidad virtual para el aprendizaje del lenguaje de señas mexicano. *Res. Comput. Sci.*, 148(8), 55-61.
- Basantes, A. V., Guerra, F. E., Naranjo, M. E., & Ibadango, D. K. (2018). Screen readers: Technological tools for the educational inclusion of blind people. *Informacion Tecnologica*, 29(5), 81-89.
- Bracamonte, O. H., López, F. R., & Navarro, R. L. (2017). Reconocimiento de patrones. *revista de investigación matemática*, 1(1).
- Bustos-Rubilar, M., Coloma, C. J., Quezada, C., Caviedes, C., Morales, M. F., Adrián, J., y País, J., (2021). La dactilología para el aprendizaje de la lectura en escolares oyentes: un estudio exploratorio. *Revista de Educación Inclusiva*, 14(1), 156-167
- CEUPE Magazine. (s.f.). Reconocimiento de patrones: qué es y cómo funciona. <https://www.ceupe.com/blog/reconocimiento-de-patrones.html>
- Cruz, M. y Serrano, J. (2014). Argumentos cuantitativos y cualitativos en sociolingüística: (Segundo Coloquio de Cambio y Variación Lingüística), 249-266.
- Cuecuecha, E., Martínez, J. J., Méndez, D., Zambrano, A., Barreto, A., Bautista, V. E., & Ayala, S. E. (2018, febrero). *Sistema de reconocimiento de vocales de la lengua de señas mexicana*. Pistas Educativas, 39(138). <http://itcelaya.edu.mx/ojs/index.php/pistas>
- Dellamary, L. E. E. (2017). La Lengua de señas mexicana, sus hablantes y su estructura. *En Comunidades Sordas en América Latina: Lengua – Cultura – Educación – Identidad*, 55-70.
- Dis-Capacidad (30 de enero del 2021). *Censo 2020: 16.5% de la población en México son personas con discapacidad*. <https://dis-capacidad.com/2021/01/30/censo-2020-16-5-de-la-poblacion-en-mexico-son-personas-con-discapacidad/>

- Daud, N. M. N, Bakar, N. A. A. A, Rusli, H. M. (2010). *Implementing rapid application development (RAD) methodology in developing practical training application system*. International Symposium on Information Technology. 1664-1666.
- Duda, R. O., Hart, P. E., & Stork D. G. (2000). *Pattern Classification*. Wiley-Interscience.
- Duda, R. O., Hart. (1973). *Pattern Classification and Scene Analysis*. Wiley-Interscience.
- Domínguez, A., (1996). Procesamiento digital de imágenes. *Perfiles Educativos*, (72).
- Economipedia. (2020). *Media geométrica*.  
<https://economipedia.com/definiciones/media-geometrica.html>
- Esparza-Maldonado, A. L., Margain-Fuentes, L. Y., Álvarez-Rodríguez, F. J., & Benítez-Guerrero, E. I. (2018). Desarrollo y evaluación de un sistema interactivo para personas con discapacidad visual. *TecnoLógicas*, 21(41), 149-157.
- Ferreira, J. A., Ferrari, F. A., Cordero, M. C., & Terminiello, L. (2017). Aplicaciones informáticas para niños con discapacidad cognitiva y/o motriz. In *IV Jornadas de Investigación, Transferencia y Extensión de la Facultad de Ingeniería (La Plata, 2017)*.
- Giacomantone, J. (2021). Análisis de imágenes y reconocimiento de patrones.
- Gómez-Terán, A., Sánchez, B. P., & Pinto, D. (2020). Postura esquelética de la mano para representar las configuraciones dactilológicas de la lengua de señas mexicana en un entorno virtual. *Res. Comput. Sci.*, 149(8), 227-240.
- González, R. C. y Woods, R. E. (2018). *Digital Image Processing*. Pearson. (4ª. Ed).
- Hernández, A. (2023). Instituto Tecnológico de Toluca contará con una ingeniería para la comunidad sorda. *El Sol de Toluca*.  
[https://www.elsoldetoluca.com.mx/local/instituto-tecnologico-de-toluca-contara-con-una-ingenieria-para-la-comunidad-sorda-9594136.html?fbclid=IwAR3u9Ok7Ma856MgzgJDFZOWhkLmk3ICt3MI5-S2pWiErlmvanxD8cdr3jsE\\_aem\\_AWGEGnwe3wBlm-\\_kbFWJxjp1\\_mzSSLZ1RUMACfb0ApfYkooQF8bCe9j2wRJJZJPf8M4](https://www.elsoldetoluca.com.mx/local/instituto-tecnologico-de-toluca-contara-con-una-ingenieria-para-la-comunidad-sorda-9594136.html?fbclid=IwAR3u9Ok7Ma856MgzgJDFZOWhkLmk3ICt3MI5-S2pWiErlmvanxD8cdr3jsE_aem_AWGEGnwe3wBlm-_kbFWJxjp1_mzSSLZ1RUMACfb0ApfYkooQF8bCe9j2wRJJZJPf8M4)
- Hernández, R., Abud, A., Rodríguez, L., Juárez, U., & Romero, C. (2018, 15 mayo). *Prototipo de intérprete de lengua de señas mexicana usando el control Leap*



Motion.149(8).[https://rcs.cic.ipn.mx/2020\\_149\\_8/Postura%20esqueletica%20de%20la%20mano%20para%20representar%20las%20configuraciones%20dactilologicas.pdf](https://rcs.cic.ipn.mx/2020_149_8/Postura%20esqueletica%20de%20la%20mano%20para%20representar%20las%20configuraciones%20dactilologicas.pdf)

Hernández, R. (2019). *Desarrollo de un intérprete de lengua de señas mexicana con vocabulario configurable según el contexto*. [Tesis de Maestría]. Instituto Tecnológico de Orizaba. <http://repositorios.orizaba.tecnm.mx:8080/xmlui/bitstream/handle/123456789/448/Tesis%20Roberto%20Hernandez%20De%20La%20Luz.pdf?sequence=1&isAllowed=y>

IBM. (s.f.). *¿Qué son las redes neuronales?* IBM Cloud. <https://www.ibm.com/es-es/topics/neural-networks>

IBM. (s.f.). *¿Qué es KNN?* IBM Cloud. <https://www.ibm.com/mx-es/topics/knn>

IFT. (2022). *Encuestas sobre sistemas operativos y navegadores 2022*. Instituto Federal de Telecomunicaciones: Usuarios y Audiencias. <https://www.ift.org.mx/usuarios-y-audiencias/encuestas-sobre-sistemas-operativos-y-navegadores-2022>

INEGI. (2020). *Presentación de Resultados. Estados Unidos Mexicanos. Censo 2020*. [https://www.inegi.org.mx/contenidos/programas/ccpv/2020/doc/Censo2020\\_Principales\\_resultados\\_EUM.pdf](https://www.inegi.org.mx/contenidos/programas/ccpv/2020/doc/Censo2020_Principales_resultados_EUM.pdf)

INSOR. (2009). *Estadísticas e información para contribuir en el mejoramiento de la calidad de vida de la población sorda colombiana*. Boletín Observatorio Social Población sorda colombiana. <http://www.insor.gov.co/historico/images/bolet%C3%ADn%20observatorio.pdf>

Interartive: A platform for contemporary art and thought. (2017). *Historia(s) de la Imagen Digital*. <https://interartive.org/2017/04/historias-de-la-imagen-digital-marisa-gomez>

Jähne, B. (2002). *Digital Image Processing*. Springer Verlag.

Karyakarte, S. y Savant, I. (2019). *International Research Journal of Engineering and Technology (IRJET). Pattern Recognition Process, Methods and Applications in Artificial Intelligence*. 6(11), 1162-1163. <https://www.irjet.net/archives/V6/i11/IRJET-V6I11121.pdf>

- Kuncheva, L. y Whitaker, C. (2014). Pattern recognition and classification. Wiley  
StatsRef: Statistics Reference Online.  
<https://lucykuncheva.co.uk/papers/lkcwWileyStatRef15.pdf>
- La Serna Palomino, N., & Román Concha, U. N. (2009). Técnicas de Segmentación en  
Procesamiento Digital de Imágenes. Revista De investigación De Sistemas E  
Informática, 6(2), 9–16.  
<https://revistasinvestigacion.unmsm.edu.pe/index.php/sistem/article/view/3299>
- Mancilla, M., Vázquez, O., Arguijo, P., Meléndez, R., Vázquez, A. (2019). Traducción  
del lenguaje de señas usando visión por computadora. Research in Computing  
Science, 148(8). [https://www.researchgate.net/profile/Pedro-Arguijo/publication/339204337\\_Traduccion\\_del\\_lenguaje\\_de\\_senas\\_usando\\_vision\\_por\\_computadora/links/5ecd4b3b45851529451450f7/Traduccion-del-lenguaje-de-senas-usando-vision-por-computadora.pdf](https://www.researchgate.net/profile/Pedro-Arguijo/publication/339204337_Traduccion_del_lenguaje_de_senas_usando_vision_por_computadora/links/5ecd4b3b45851529451450f7/Traduccion-del-lenguaje-de-senas-usando-vision-por-computadora.pdf)
- Marín Díaz, V. (2018). La realidad aumentada al servicio de la inclusión educativa.  
Estudio de caso.
- MathWorks. (s.f.). Reconocimiento de patrones (Pattern Recognition).  
<https://la.mathworks.com/discovery/pattern-recognition.html>
- Moreno, J. C. (2018). *Lenguas de señas, historia de la lingüística y lingüística general*.  
Actas do XIII Congreso Internacional de Lingüística Xeral, 23(30).  
<http://cilx2018.uvigo.gal/actas/pdf/plen02.pdf>
- Narasimha, M. y Susheela, D. (2015). Introduction to Pattern Recognition and Machine  
Learning. IISc Lecture Notes Series.
- O'Handley, D. y Green, W. (1972). Recent developments in digital image processing at  
the image processing laboratory at the jet propulsion laboratory. Proceedings of  
the IEEE. 60(7), 821-828. <https://www.semanticscholar.org/paper/Recent-developments-in-digital-image-processing-at-O'Handley-Green/96a41fcf864b13f955708509c7d354374292d6ea>
- OpenCV - Open Source Computer Vision. (s.f.). Structural Analysis and Shape  
Descriptors.[https://docs.opencv.org/3.4/d3/dc0/group\\_imgproc\\_shape.html#gaf2b97a230b51856d09a2d934b78c015fa73b8cbe851905080a1d918c902253d](https://docs.opencv.org/3.4/d3/dc0/group_imgproc_shape.html#gaf2b97a230b51856d09a2d934b78c015fa73b8cbe851905080a1d918c902253d)  
[cc](https://creativecommons.org/licenses/by/4.0/)

- Paolanti, M., & Frontoni, E. (2020). Multidisciplinary pattern recognition applications: A review. *Computer Science Review*, 37, 100276.
- ProbabilidadyEstadistica.net. (s.f.). *Coeficiente de correlación de Pearson*. <https://www.probabilidadyestadistica.net/coeficiente-de-correlacion-de-pearson/>
- Python Wiki. (2023). GUI Programming in Python. <https://wiki.python.org/moin/GuiProgramming>
- Quevedo, J.P., Pupo, J. R., Godínez, J. C. & Uribe, R. J. (2019). *Estrategia integral para la inclusión con las personas en condición de discapacidad auditiva, una apuesta en el marco de la educación*. Experiencia Docente.
- Rivas Asanza, W., Masón Olivo, B., Tusa, E. (2018). Tecnologías utilizadas en el reconocimiento de patrones y clasificación de Imágenes. Machala: Universidad Técnica de Machala.
- Robins, R. H. (1964). *General Linguistics: An Introductory Survey*. Longman.
- Sánchez, B. H., & Cedeño, G. G. (2021). Uso y presencia de las tecnologías en personas con discapacidad. In *Congreso Nacional de Ciencia y Tecnología–APANAC*, pp. 329-335.
- Santa Cruz, C., Espinoza, V., & Hohlberg, E. (2021). *Problemas Socioemocionales en Niños con Discapacidad Auditiva, Discapacidad Visual y Desarrollo Típico*. Revista Latinoamericana de Educación Inclusiva, 15(1). <http://www.rinace.net/rlei/numeros/vol15-num1/art7.pdf>
- Serafín de Fleischmann, M. E. y González, P. R. (2011). *Manos con voz. Diccionario de Lenguas de Señas Mexicana*. Conapred.
- Sistema Nacional DIF (2017). *¿Qué es la discapacidad auditiva?* <https://www.gob.mx/difnacional/articulos/que-es-la-discapacidad-auditiva>
- Sharma, P. y Kaur, M. (2013). International Journal of Advanced Research in Computer Science and Software Engineering. Classification in Pattern Recognition: A Review. 3(4), 301. <http://lapi.fi-p.unam.mx/wp-content/uploads/Pattern-recognition-review.pdf>

Trujillo-Romero, F., y García Bautista, G. (2021). Reconocimiento de palabras de la Lengua de Señas Mexicana utilizando información RGB-D. ReCIBE, Revista electrónica De Computación, Informática, Biomédica Y Electrónica, 10(2), C2–23. <https://doi.org/10.32870/recibe.v10i2.209>

## APÉNDICE A

### Base de Conocimientos

**Figura A.1**

*Letra "A"*



**Figura A.2**

*Letra "B"*



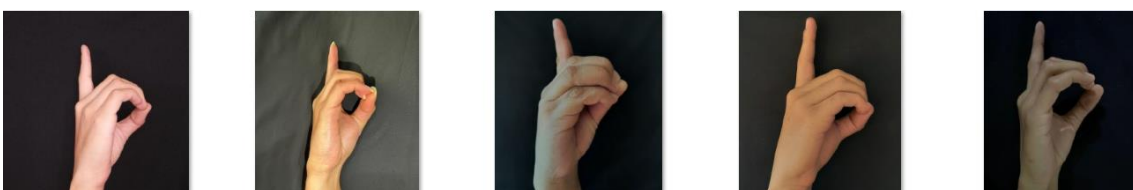
**Figura A.3**

*Letra "C"*



**Figura A.4**

*Letra "D"*



**Figura A.5**

*Letra "E"*



**Figura A.6**

*Letra "F"*



**Figura A.7**

*Letra "G"*



**Figura A.8**

*Letra "H"*



**Figura A.9**

*Letra "I"*



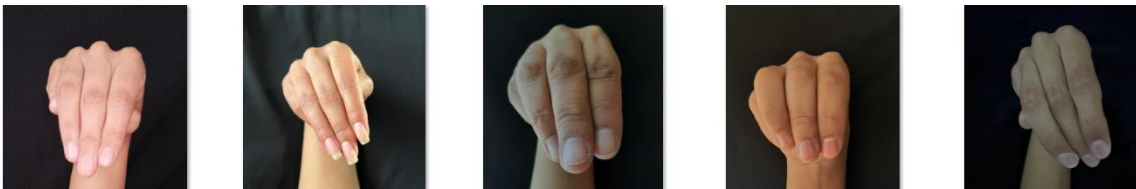
**Figura A.10**

*Letra "L"*



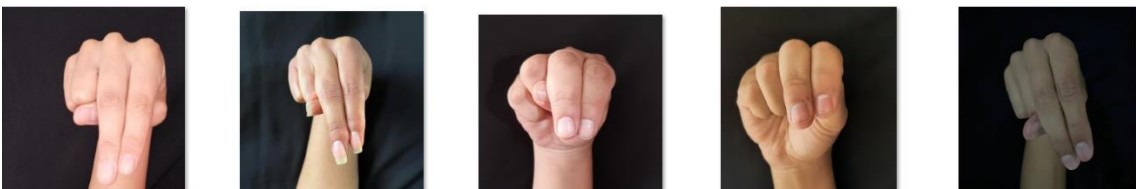
**Figura A.11**

*Letra "M"*



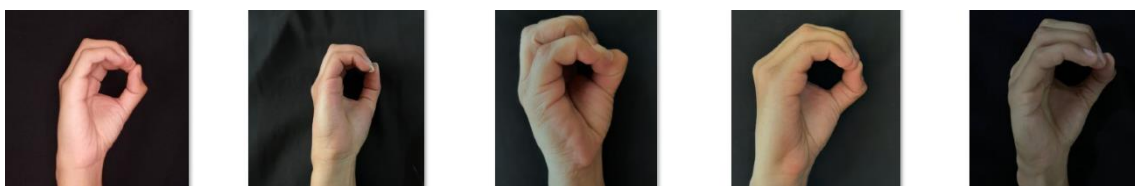
**Figura A.12**

*Letra "N"*



**Figura A.13**

*Letra "O"*



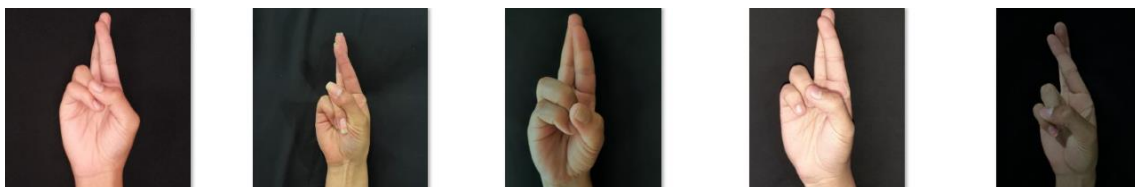
**Figura A.14**

*Letra "P"*



**Figura A.15**

*Letra "R"*



**Figura A.16**

*Letra "S"*





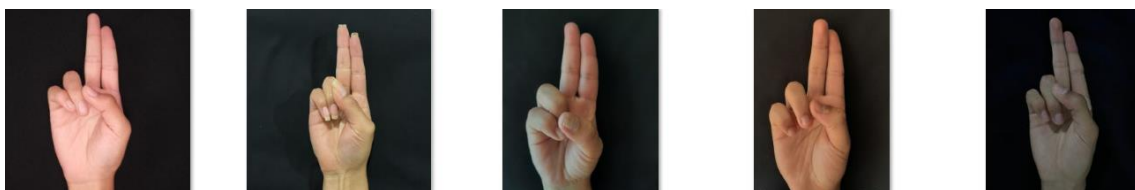
**Figura A.17**

*Letra "T"*



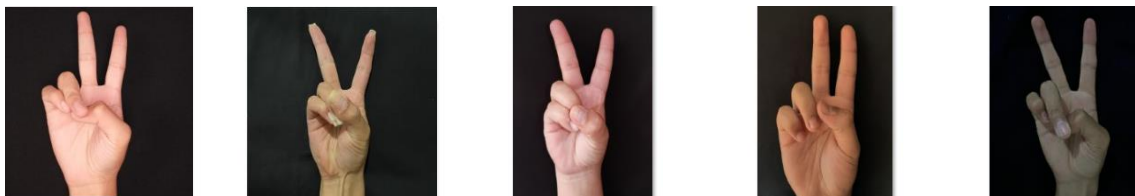
**Figura A.18**

*Letra "U"*



**Figura A.19**

*Letra "V"*



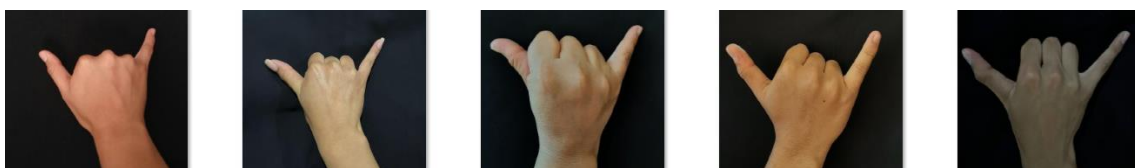
**Figura A.20**

*Letra "W"*



**Figura A.21**

*Letra "Y"*



## APÉNDICE B

**Figura B.1**

*Muestra de las tabla de distancias entre las señas a identificar contra las señas  
contenida en la base de conocimiento.*

	CC	CD	CE	CF	CG	CH	CI	CJ	CK	CL	CM	CN	CO	CP	CQ	CR	CS	CT	CU	CV	CW	CX	CY	CZ	DA	DB	DC	DD
1	S <sub>5</sub>	T <sub>1</sub>	T <sub>2</sub>	T <sub>3</sub>	T <sub>4</sub>	T <sub>5</sub>	U <sub>1</sub>	U <sub>2</sub>	U <sub>3</sub>	U <sub>4</sub>	U <sub>5</sub>	V <sub>1</sub>	V <sub>2</sub>	V <sub>3</sub>	V <sub>4</sub>	V <sub>5</sub>	W <sub>1</sub>	W <sub>2</sub>	W <sub>3</sub>	W <sub>4</sub>	W <sub>5</sub>	Y <sub>1</sub>	Y <sub>2</sub>	Y <sub>3</sub>	Y <sub>4</sub>	Y <sub>5</sub>	LETRA	MEDIA
2	1.45	1.714	1.375	1.442	1.559	1.952	1.745	1.954	1.777	1.703	2.028	1.927	3.411	3.418	1.807	3.119	2.156	1.908	1.732	1.999	2.126	0.211	1.534	0.478	1.289	1.45	A	0.406035662
3	1.004	1.268	0.924	0.996	1.113	1.506	1.581	1.79	1.613	1.727	1.864	1.764	3.247	3.254	1.653	2.955	1.992	1.744	1.568	1.835	1.962	0.289	1.35	0.793	1.736	1.59		
4	1.265	1.516	1.226	1.247	1.371	1.787	1.881	2.09	1.914	1.995	2.164	2.064	3.547	3.554	1.944	3.255	2.292	2.045	1.868	2.136	2.262	0.463	1.643	0.533	1.487	1.37		
5	1.833	2.097	1.753	1.825	1.942	2.335	10.83	10.92	10.7	10.54	10.86	11	12.05	12.36	10.62	11.76	11.35	11.37	1.949	11.51	11.22	0.257	1.64	0.276	1.145	1.4		
6	1.164	1.409	1.379	1.145	1.269	1.685	2.067	2.276	2.312	2.441	2.35	2.25	3.733	3.74	2.367	3.441	2.478	2.231	2.054	2.322	2.448	0.765	1.801	0.511	1.609	1.21		
7	1.287	1.189	1.327	1.401	1.386	1.144	2.306	2.545	2.591	2.81	2.575	2.243	1.741	1.748	2.792	1.949	1.731	1.575	1.195	1.556	2.148	2.132	2.495	2.659	3.943	3.54		
8	1.785	1.539	1.824	1.803	1.68	1.264	2.803	2.9	3.004	3.294	2.93	2.728	2.06	1.787	3.271	2.305	2.194	1.941	1.653	1.911	2.503	2.362	2.456	2.889	4.44	4.04		
9	2.079	2.087	1.943	2.326	2.257	1.928	2.091	2.285	2.261	2.371	2.212	1.974	1.144	0.89	2.312	1.384	1.638	1.75	1.579	1.721	2.066	3.105	2.991	3.632	4.356	4.28	B	0.644725872
10	0.89	1.037	0.929	1.25	1.235	0.993	2.034	2.393	2.439	2.568	2.423	1.895	1.892	1.9	2.494	1.798	1.438	1.423	1.044	1.404	1.996	1.858	2.382	2.385	3.545	3.14		
11	1.093	1.356	0.781	1.569	1.554	1.312	2.165	2.272	2.519	2.899	2.435	2.095	2.642	2.343	2.901	2.359	1.429	0.903	0.368	0.784	1.749	1.833	2.662	2.36	3.176	3.24		
12	1.057	0.941	1.092	0.749	0.686	0.779	1.598	1.894	1.962	2.118	1.924	1.438	2.831	2.838	2.039	2.539	1.576	1.328	1.152	1.419	1.546	1.191	1.405	1.718	3.15	2.75		
13	1.062	1.2	0.922	0.983	0.974	1.074	0.481	0.316	0.562	0.747	0.344	0.352	0.965	1.219	0.683	0.725	0.614	0.358	0.53	0.388	0.071	1.498	1.384	2.025	2.749	2.67		
14	0.499	0.637	0.359	0.673	0.603	0.511	0.498	0.739	0.749	0.793	0.671	0.306	0.887	1.221	0.674	0.59	0.098	0.204	0.156	0.201	0.52	1.452	1.338	1.979	2.702	2.62	C	0.465616975
15	1.044	1.307	0.732	1.52	1.505	1.263	2.32	2.704	2.838	3.079	2.88	2.25	2.845	2.545	3.056	2.552	1.559	1.238	0.474	1.17	2.182	1.809	2.613	2.336	3.177	3.19		
16	1.483	1.746	1.171	1.959	1.944	1.702	1.963	2.061	2.316	2.697	2.245	1.893	2.27	1.971	2.699	1.986	1.202	0.635	0.496	0.517	1.487	1.918	3.052	2.445	3.381	3.63		
17	1.877	2.27	1.561	2.027	2.072	2.259	1	1.205	1.029	0.955	1.279	1.179	2.662	2.669	1.059	2.37	1.407	1.16	1.35	1.251	1.377	0.542	1.574	0.973	1.902	2.15		
18	1.954	2.346	1.638	2.103	2.149	2.335	1.076	0.92	1.188	1.569	1.225	1.338	2.731	2.39	1.571	2.481	1.323	0.664	1.012	0.731	0.953	1.144	2.35	1.671	2.678	2.93		
19	1.771	2.164	1.455	1.92	1.966	2.153	0.95	1.009	0.847	0.866	1.083	1.207	2.466	2.473	1.02	2.174	1.328	0.963	0.875	1.054	1.181	0.773	1.77	1.3	2.099	2.35		
20	1.881	2.274	1.565	2.031	2.076	2.263	9.79	9.869	9.65	9.492	9.814	9.955	11	11.31	9.57	10.71	10.29	10.32	1.387	10.46	10.17	0.58	1.468	1.011	1.797	2.04		
21	1.619	2.011	1.303	1.768	1.814	2.001	1.658	1.958	1.825	1.951	2.162	1.92	3.742	3.442	1.997	3.449	1.96	1.417	0.849	1.459	2.005	0.794	1.786	1.321	2.115	2.36		
22	0.258	0.548	0.16	0.734	0.719	0.537	1.777	2.136	2.182	2.31	2.166	1.628	2.408	2.416	2.236	2.116	1.18	1.165	0.786	1.147	1.738	1.23	1.957	1.711	2.74	2.47		