

**UNIVERSIDAD NACIONAL DEL SANTA**

**FACULTAD DE INGENIERÍA**

**Escuela Profesional de Ingeniería de Sistemas e Informática**



**UNS**  
UNIVERSIDAD  
NACIONAL DEL SANTA

**VISIÓN COMPUTACIONAL PARA LA TRADUCCIÓN EN  
TIEMPO REAL DEL LENGUAJE DE SEÑAS A TEXTO EN  
IDIOMA ESPAÑOL**

**Tesis para Optar el Título de Ingeniero de Sistemas e Informática**

**Tesistas:**

**Bach. Eder Alberto Albino Huertas**

**Bach. Leticia Teresa López Olivos**

**Asesor:**

**Mg. Luis Enrique Ramírez Milla**

**NUEVO CHIMBOTE – PERÚ**

**2018**



**UNS**  
UNIVERSIDAD  
NACIONAL DEL SANTA

## **FACULTAD DE INGENIERIA**

**ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS E INFORMÁTICA**

### **ACTA DE EVALUACIÓN PARA SUSTENTACIÓN DE TESIS**

En el Campus Universitario de la Universidad Nacional del Santa, siendo las 4:30 p.m. del día lunes 02 de julio de 2018, en el Aula S3 del Pabellón nuevo de la EPISI, en atención a la Resolución Decanal N° 349-2018-UNS-FI de Declaración de Expedito de fecha 26.06.18; se llevó a cabo la instalación del jurado Evaluador, designado mediante Resolución N° 177 - 2018 -UNS-CFI de fecha 24.05.2018, integrado por el **Dr. Juan Pablo Sánchez Chávez (Presidente)**, **Ms. Camilo Suárez Rebaza (Secretario)**, **Mg. Luis Enrique Ramírez Milla (Integrante)**, para dar inicio a la sustentación del Informe Final de Tesis, cuyo título es: **"VISIÓN COMPUTACIONAL PARA LA TRADUCCIÓN EN TIEMPO REAL DEL LENGUAJE DE SEÑAS A TEXTO EN IDIOMA ESPAÑOL"**, perteneciente al bachiller: **EDER ALBERTO ALBINO HUERTAS** con código de matrícula N° **201214015**, tiene como **ASESOR** al **Mg. LUIS ENRIQUE RAMÍREZ MILLA**, según **T/R.D. N° 630- 2017 -UNS-FI** de fecha 15.09.2017

Terminada la sustentación, el tesista respondió a las preguntas formuladas por los miembros del Jurado Evaluador y el público presente.

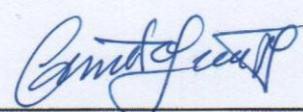
El Jurado después de deliberar sobre aspectos relacionados con el trabajo, contenido y sustentación del mismo y con las sugerencias pertinentes y en concordancia con el artículo 40º del Reglamento General para Obtener el Grados Académico de Bachiller y el Títulos Profesional en la Universidad Nacional del Santa; considera la siguiente nota final de Evaluación:

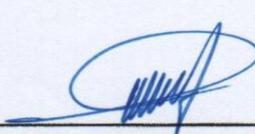
BACHILLER	PROMEDIO	PONDERACIÓN
<b>EDER ALBERTO ALBINO HUERTAS</b>	18	BUENO

Siendo la 5: 30 p.m. se dio por terminado el Acto de Sustentación y en señal de conformidad, firma el Jurado la presente Acta.

Nuevo Chimbote, 02 de julio de 2018

  
\_\_\_\_\_  
**DR. JUAN PABLO SÁNCHEZ CHÁVEZ**  
**PRESIDENTE**

  
\_\_\_\_\_  
**MS. CAMILO SUÁREZ REBAZA**  
**SECRETARIO**

  
\_\_\_\_\_  
**MG. LUIS ENRIQUE RAMÍREZ MILLA**  
**INTEGRANTE**



**UNS**  
UNIVERSIDAD  
NACIONAL DEL SANTA

## **FACULTAD DE INGENIERIA**

**ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS E INFORMÁTICA**

### **ACTA DE EVALUACIÓN PARA SUSTENTACIÓN DE TESIS**

En el Campus Universitario de la Universidad Nacional del Santa, siendo las 4:30 p.m. del día lunes 02 de julio de 2018, en el Aula S3 del Pabellón nuevo de la EPISI, en atención a la Resolución Decanal N° 349-2018-UNS-FI de Declaración de Expedito de fecha 26.06.18; se llevó a cabo la instalación del jurado Evaluador, designado mediante Resolución N° 177 - 2018 -UNS-CFI de fecha 24.05.2018, integrado por el **Dr. Juan Pablo Sánchez Chávez (Presidente)**, **Ms. Camilo Suárez Rebaza (Secretario)**, **Mg. Luis Enrique Ramírez Milla (Integrante)**, para dar inicio a la sustentación del Informe Final de Tesis, cuyo título es: **"VISIÓN COMPUTACIONAL PARA LA TRADUCCIÓN EN TIEMPO REAL DEL LENGUAJE DE SEÑAS A TEXTO EN IDIOMA ESPAÑOL"**, perteneciente a la bachiller: **LETICIA TERESA LÓPEZ OLIVOS** con código de matrícula N° 201214037, tiene como **ASESOR** al **Mg. LUIS ENRIQUE RAMÍREZ MILLA**, según **T/R.D. N° 630- 2017 -UNS-FI** de fecha 15.09.2017

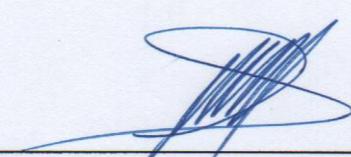
Terminada la sustentación, la tesista respondió a las preguntas formuladas por los miembros del Jurado Evaluador y el público presente.

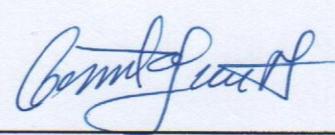
El Jurado después de deliberar sobre aspectos relacionados con el trabajo, contenido y sustentación del mismo y con las sugerencias pertinentes y en concordancia con el artículo 40º del Reglamento General para Obtener el Grados Académico de Bachiller y el Títulos Profesional en la Universidad Nacional del Santa; considera la siguiente nota final de Evaluación:

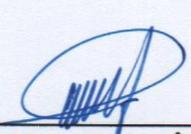
BACHILLER	PROMEDIO	PONDERACIÓN
<b>LETICIA TERESA LÓPEZ OLIVOS</b>	18	BUENO

Siendo la 5: 30 p.m. se dio por terminado el Acto de Sustentación y en señal de conformidad, firma el Jurado la presente Acta.

Nuevo Chimbote, 02 de julio de 2018

  
\_\_\_\_\_  
**DR. JUAN PABLO SÁNCHEZ CHÁVEZ**  
**PRESIDENTE**

  
\_\_\_\_\_  
**MS. CAMILO SUÁREZ REBAZA**  
**SECRETARIO**

  
\_\_\_\_\_  
**MG. LUIS ENRIQUE RAMÍREZ MILLA**  
**INTEGRANTE**

**UNIVERSIDAD NACIONAL DEL SANTA**

**FACULTAD DE INGENIERÍA**

**Escuela Profesional de Ingeniería de Sistemas e Informática**

**TÍTULO**

**VISIÓN COMPUTACIONAL PARA LA TRADUCCIÓN EN  
TIEMPO REAL DEL LENGUAJE DE SEÑAS A TEXTO EN  
IDIOMA ESPAÑOL**

**Tesis para Optar el Título de Ingeniero de Sistemas e Informática**

**REVISADO Y APROBADO POR:**

---

**Mg. Luis Enrique Ramírez Milla**

**Asesor**

**NUEVO CHIMBOTE-PERU**

**2018**

**UNIVERSIDAD NACIONAL DEL SANTA**

**FACULTAD DE INGENIERÍA**

**Escuela Profesional De Ingeniería De Sistemas E Informática**

**TÍTULO**

**VISIÓN COMPUTACIONAL PARA LA TRADUCCIÓN EN  
TIEMPO REAL DEL LENGUAJE DE SEÑAS A TEXTO EN  
IDIOMA ESPAÑOL**

**Tesis para Optar el Título de Ingeniero de Sistemas e Informática**

**REVISADO Y APROBADO POR EL JURADO EVALUADOR:**

\_\_\_\_\_  
**Dr. Juan Pablo Sánchez Chávez**  
**Presidente**

\_\_\_\_\_  
**Ms. Camilo Suárez Rebaza**  
**Secretario**

\_\_\_\_\_  
**Mg. Luis Enrique Ramírez Milla**  
**Integrante**

\_\_\_\_\_  
**Ms. Dayan Fernando Macedo Alcántara**  
**Accesitario**

**NUEVO CHIMBOTE-PERU**

**2018**

## DEDICATORIA

A Dios, que protege y guía nuestros pasos para superar las adversidades,  
manteniendo en nuestros corazones la fe y la esperanza.

A nuestras Madres, Sra. Yolanda Huertas Murga y Sra. Teresa Aurora Olivos Pérez, quienes, con su amor, su ternura, su dedicación, su entrega y su lucha constante, nos inculcaron el deseo de superación, de ser mejores personas cada día y de no rendirnos ante las dificultades A nuestras hermanas, Sandy Daniela Albino Huertas y Almendra Rosita Castro Olivos, por su apoyo incondicional, sus consejos y su compañía que alegra nuestras vidas.

A nuestros Amigos y Compañeros: Carlos, Esteban, Pablo, Max, Guillermo, Yorka, Fredy, Eduardo, Bryan, a todos, quienes estuvieron presentes cuando más los necesitábamos y que siempre nos alentaron a seguir adelante.

Bach. Eder Alberto Albino Huertas.

Bach. Leticia Teresa López Olivos.

## **AGRADECIMIENTO**

A Dios, por darnos la Vida, así como la sabiduría y las fuerzas necesarias para alcanzar las metas que nos hemos propuesto.

A nuestras familias, por su apoyo incondicional, tanto moral como económico, por su comprensión y su confianza, por su sacrificio constante y su esfuerzo, siendo los impulsores en nuestro desarrollo y formación profesional. Muchas gracias por su infinito amor.

A los Docentes de: I.E.P. “San José”, I.E. N°89002 “Gloriosa Ex 329”, I.E. María Reiche Newmann-Jaihua, CEPUNS, Universidad Nacional del Santa. Quienes nos brindaron sus conocimientos, orientaron y confiaron en nosotros, lo que forjó nuestras capacidades como estudiantes y como profesionales. Gracias por saber escuchar y ser más que docentes y convertirse en amigos.

A la doctora Yvonne del Rocío Angulo Serrano y a la Clínica Dental VITALDENT Chimbote, por haberme dado la oportunidad de realizar mis prácticas pre-profesionales en su establecimiento, así como su apoyo y orientación.

Al Centro de Educación Básica Especial N° 01, por abrirnos sus puertas y recibirnos con cariño y compromiso, brindándonos las facilidades que permitieron la culminación exitosa de nuestra investigación.

Bach. Eder Alberto Albino Huertas.

Bach. Leticia Teresa López Olivos.

# ÍNDICE

<b>DEDICATORIA</b> .....	vi
<b>AGRADECIMIENTO</b> .....	vii
<b>RESUMEN</b> .....	xiii
<b>ABSTRACT</b> .....	xiv
<b>PRESENTACIÓN</b> .....	xv
<b>INTRODUCCIÓN</b> .....	1
<b>DATOS GENERALES DEL ESTUDIO</b> .....	3
<b>CAPÍTULO I: LA INSTITUCIÓN</b> .....	6
<b>1.1. Descripción</b> .....	6
<b>1.2. Direccionamiento Estratégico</b> .....	9
<b>1.3. Objetivos de la Institución</b> .....	11
<b>CAPÍTULO II: PLANTEAMIENTO DEL PROBLEMA</b> .....	13
<b>2.1. Problema</b> .....	13
<b>2.2. Antecedentes</b> .....	16
<b>2.3. Alcance</b> .....	21
<b>2.4. Importancia de la investigación</b> .....	22
<b>2.5. Formulación del problema</b> .....	22
<b>2.6. Formulación de hipótesis</b> .....	22
<b>2.7. Objetivos de la investigación</b> .....	23
<b>CAPÍTULO III: MARCO TEÓRICO</b> .....	24
<b>3.1. Marco Conceptual</b> .....	24
<b>3.2. Marco Teórico</b> .....	29
<b>CAPÍTULO IV: METODOLOGÍA DE DESARROLLO</b> .....	42
<b>4.1. Aplicación de la Metodología SCRUM</b> .....	42
<b>4.2. Definición de Parámetros</b> .....	54
<b>4.3. Reconocimiento de patrones</b> .....	60
<b>4.4. Procesamiento de imágenes</b> .....	63
<b>4.5. Actualización del Sprint Backlog</b> .....	78
<b>4.6. Traducción del Lenguaje de Señas</b> .....	85

<b>CAPÍTULO V: MATERIALES Y MÉTODOS .....</b>	<b>96</b>
<b>5.1. Método de Investigación .....</b>	<b>96</b>
<b>CAPÍTULO VI: RESULTADOS Y DISCUSIONES .....</b>	<b>101</b>
<b>CONCLUSIONES.....</b>	<b>118</b>
<b>RECOMENDACIONES.....</b>	<b>121</b>
<b>REFERENCIAS BIBLIOGRÁFICAS .....</b>	<b>122</b>
<b>ANEXOS .....</b>	<b>124</b>

## ÍNDICE DE ILUSTRACIONES

Ilustración N° 1: Ubicación Geográfica CEBE N° 01 .....	7
Ilustración N° 2: Organigrama Estructural del CEBE N° 01 .....	9
Ilustración N° 3: Etapas de un sistema de visión artificial.....	25
Ilustración N° 4: Diagrama de bloques de un Sistema de Visión Computacional .....	27
Ilustración N° 5: Lenguaje de señas .....	27
Ilustración N° 6: Sistema dactilológico peruano (español).....	28
Ilustración N° 7: Ejemplo del Algoritmo K-vecino más cercano .....	31
Ilustración N° 8: Representación de un kernel y el movimiento que realiza para recorrer toda la imagen. ....	34
Ilustración N° 9: Imagen junto a su correspondiente histograma. ....	34
Ilustración N° 10: Imagen junto al resultado de su umbralización con $U=60$ . ....	35
Ilustración N° 11: Proceso de segmentación automática de la imagen .....	36
Ilustración N° 12: Dilatación binaria Máscara B. ....	37
Ilustración N° 13: Dilatación binaria Imagen A.....	37
Ilustración N° 14: Imagen resultante $A \oplus B$ .....	37
Ilustración N° 15: Imagen binarizada y dilatación de la misma (derecha). ....	38
Ilustración N° 16: Erosión binaria Máscara B .....	38
Ilustración N° 17: Erosión binaria Imagen A.....	39
Ilustración N° 18: Imagen resultante $A \ominus B$ .....	39
Ilustración N° 19: Imagen binarizada (izquierda) y erosionada (derecha).....	39
Ilustración N° 20: Gráfico de valores de Story Points .....	46
Ilustración N° 21: Gráfico de Duración del Sprint.....	49
Ilustración N° 22: Total de Sprints obtenidos. ....	52
Ilustración N° 24: Letras dobles del alfabeto dactilológico peruano .....	54
Ilustración N° 25: Alfabeto seleccionado del Lenguaje de Señas Peruano.....	56
Ilustración N° 26: Número de personas para la obtención de imágenes .....	56
Ilustración N° 27: Videos de la representación del alfabeto seleccionado.....	57
Ilustración N° 28: Imágenes por cada letra. ....	57
Ilustración N° 29: Ejemplo 1- Correlación de imágenes.....	61
Ilustración N° 30: Ejemplo 2- Imagen general.....	61
Ilustración N° 31: Ejemplo 2- Patrón.....	62
Ilustración N° 32: Ejemplo 2- Patrón localizado por correlación. ....	62
Ilustración N° 33: Aplicación de filtro de Media. ....	63
Ilustración N° 34: Aplicación de filtro Gaussiano. ....	64
Ilustración N° 35: Aplicación de filtro de Media. ....	64
Ilustración N° 36: Aplicación de filtro Bilateral. ....	65
Ilustración N° 37: Aplicación de balanceo de contraste.....	66
Ilustración N° 38: Matriz para operador de media. ....	67
Ilustración N° 39: Matrices de operador laplaciana. ....	67
Ilustración N° 40: Aplicación del realce de bordes con umbral de operador laplaciana. ....	68
Ilustración N° 41: Aplicación del Algoritmo Canny.....	70
Ilustración N° 42: Matrices de operador sobel.....	70
Ilustración N° 43: Aplicación de Algoritmo Sobel. ....	71
Ilustración N° 44: Ejemplo 1- Segmentación con umbrales .....	72

Ilustración N° 45: Ejemplo 2 - Segmentación con umbrales .....	73
Ilustración N° 46: Ejemplo de erosión. ....	75
Ilustración N° 47: Ejemplo de dilatación. ....	76
Ilustración N° 48: Ejemplo de apertura.....	77
Ilustración N° 49: Ejemplo de cierre.....	77
Ilustración N° 50: Aplicación de umbral en matriz de imagen. ....	81
Ilustración N° 51: Aplicación de rango de valores. ....	82
Ilustración N° 52: Aplicación del espacio de color YCr-CB. ....	83
Ilustración N° 53: Eliminación de ruido. ....	83
Ilustración N° 54: Rango de color.....	84
Ilustración N° 55: Imagen binarizada. ....	84
Ilustración N° 56: Conversión a Escala de Grises.....	86
Ilustración N° 57: Gradiente en dirección Horizontal (eje x) .....	87
Ilustración N° 58: Gradiente en dirección Vertical (eje y).....	87
Ilustración N° 59: Fusión de gradientes. ....	87
Ilustración N° 60: División en celdas.....	88
Ilustración N° 61: Cálculo del histograma de orientación .....	88
Ilustración N° 62: Mockup de la interfaz del modelo .....	89
Ilustración N° 63: Interfaz real de modelo de reconocimiento .....	90
Ilustración N° 64: Ejemplo ilustrativo del Clasificador SVM .....	91
Ilustración N° 65: Etiquetado de imágenes con las letras correspondientes .....	92
Ilustración N° 66: Proceso de selección.....	93
Ilustración N° 67: Pruebas de reconocimiento.....	94
Ilustración N° 68: Infografía del Modelo de Visión Computacional .....	95
Ilustración N° 23: Planeación de la Investigación. ....	100
Ilustración N° 69: Resultados de la Pregunta N°1 de la encuesta .....	116
Ilustración N° 70: Resultados de la Pregunta N°2 de la encuesta .....	116
Ilustración N° 71: Resultados de la Pregunta N°3 de la encuesta.....	117

## ÍNDICE DE TABLAS

Tabla N° 1: Indicadores de las variables del proyecto.....	23
Tabla N° 2: Complejidad de Requerimientos .....	45
Tabla N° 3: Story Points para cada User Story Funcional. ....	47
Tabla N° 4: Story Points para cada User Story No Funcional. ....	47
Tabla N° 5: User Story Funcionales con puntuación mayor a 20 .....	48
Tabla N° 6: User Story No Funcionales con puntuación mayor a 20 .....	48
Tabla N° 7: Cronograma de actividades. ....	53
Tabla N° 8: Porcentaje de aciertos en el proceso de traducción .....	104
Tabla N° 9: Comparación de Varianzas de los aciertos .....	107
Tabla N° 10: Comparación de Medias de los aciertos .....	108
Tabla N° 11: Tiempo en cada prueba del proceso de traducción.....	110
Tabla N° 12: Comparación de Varianzas del tiempo de traducción .....	113
Tabla N° 13: Comparación de Medias del tiempo de traducción.....	114

## RESUMEN

El presente trabajo de investigación tiene como objetivo general traducir en tiempo real las principales posiciones del lenguaje de señas a texto en idioma español, para esto se trabajará con la librería de visión computacional OpenCV la cual es muy robusta e idónea para la investigación además para la creación del prototipo se tienen como procesos determinantes la binarización, filtros y segmentación, los cuales permitirán la extracción de características particulares de las imágenes, por último para evaluar el performance del prototipo se tendrán los indicadores de velocidad y precisión en la traducción del lenguaje de señas. El resultado positivo de esta investigación tendrá un impacto importante en las personas con limitaciones auditivas o del habla ya que podrán comunicarse con el resto de personas que desconocen el lenguaje de señas.

Investigadores:

- Bach. Eder Alberto Albino Huertas.
- Bach. Leticia Teresa López Olivos.

Asesor:

- Mg. Luis Enrique Ramírez Milla.

## **ABSTRACT**

The main objective of this research work is to translate the main positions of sign language into text in real time in Spanish, for this purpose we will work with the OpenCV computational vision library, which is very robust and suitable for research, as well as for creation of the prototype are binarization processes, filters and segmentation, which will allow the extraction of particular characteristics of the images, finally to evaluate the performance of the prototype will be the indicators of speed and accuracy in the translation of sign language . The positive result of this research will have an important impact on people with hearing or speech limitations since they will be able to communicate with the rest of people who do not know sign language.

Researchers:

- Bach. Eder Alberto Albino Huertas.
- Bach. Leticia Teresa López Olivos.

Adviser:

- Mg. Luis Enrique Ramírez Milla.

## **PRESENTACIÓN**

### **Señores miembros del jurado:**

En cumplimiento a lo dispuesto en el Reglamento General de Grados y Títulos de la Universidad Nacional del Santa, ponemos a vuestra consideración el presente Proyecto de Tesis titulado: **“VISIÓN COMPUTACIONAL PARA LA TRADUCCIÓN EN TIEMPO REAL DEL LENGUAJE DE SEÑAS A TEXTO EN IDIOMA ESPAÑOL”**, con la finalidad de optar el Título de Ingeniero de Sistemas e Informática.

Este trabajo nació con la intención de lograr evidenciar la inclusión de las personas sordomudas a la sociedad, contribuyendo a que puedan comunicarse y desenvolverse más fácilmente, esto tendrá como resultado el aumento de oportunidades de superación para estas personas con capacidades diferentes. Esta investigación tiene como propósito la traducción en tiempo real del lenguaje de señas a texto en idioma español, la cual servirá como base para futuras investigaciones que se puedan realizar; de esta manera estamos ayudando a romper barreras de discriminación en nuestra sociedad.

Por todo lo expuesto, a ustedes señores miembros del jurado evaluador, ponemos a su disposición este Proyecto de Tesis, para su revisión, esperando cumpla con los requisitos para su aprobación.

Atentamente

Bach. Eder Alberto Albino Huertas.

Bach. Leticia Teresa López Olivos.

# INTRODUCCIÓN

La presente investigación describe cada uno de los elementos y los procesos necesarios para lograr la traducción en tiempo real del Lenguaje de Señas a texto en idioma español, a través de la Visión Computacional.

Con el objetivo de que la investigación permita definir los parámetros exactos para la traducción del lenguaje empleado por personas con limitaciones auditivas o del habla, se analizan los diferentes algoritmos de reconocimiento de patrones y se comprueba la efectividad de los mismos

El modelo de Visión Computacional es una herramienta que procesa, analiza y clasifica imágenes en tiempo real, facilitando la comunicación entre las personas sordomudas y su entorno, y contribuyendo a reducir las brechas de discriminación que existe en la sociedad.

Este trabajo está dividido en seis capítulos, cada uno estructurado de forma clara y precisa con la finalidad de que el proyecto sea entendible y coherente para el lector, estos capítulos se describen a continuación:

- El Capítulo I detalla la información obtenida acerca de la institución donde se llevó a cabo las entrevistas y pruebas del proyecto.
  
- El Capítulo II se basa en el planteamiento del problema, especificando la Realidad Problemática, Análisis del Problema, Formulación del Problema, Antecedentes, Justificación y Objetivos.

- El Capítulo III detalla el Fundamento Teórico, el cual es dividido en Marco Conceptual y Marco Teórico; el primero abarca los conceptos básicos sobre el proyecto y el segundo contiene la teoría acerca de la metodología de análisis y diseño empleada en este trabajo.
  
- El Capítulo IV determina los roles del equipo, la duración del trabajo y el desarrollo del modelo de Visión Computacional, delimitando el área de investigación y el entorno tecnológico donde se va a trabajar, los cuales representan la base para el desarrollo de la metodología SCRUM.
  
- El Capítulo V establece los materiales y métodos que se van a emplear para la culminación exitosa de la investigación.
  
- El Capítulo VI plasma los resultados obtenidos en las pruebas realizadas a la muestra de población determinada para el desarrollo de este proyecto.
  
- Finalmente se da a conocer las conclusiones del proyecto y las recomendaciones para futuras investigaciones relacionadas con el tema de estudio.

## **DATOS GENERALES DEL ESTUDIO**

❖ **Título**

“VISIÓN COMPUTACIONAL PARA LA TRADUCCIÓN EN TIEMPO REAL DEL LENGUAJE DE SEÑAS A TEXTO EN IDIOMA ESPAÑOL”

❖ **Tesistas**

Bach. Albino Huertas Eder Alberto

Bach. López Olivos Leticia Teresa

❖ **Asesor**

Mg. Luis Enrique Ramírez Milla.

❖ **Tipo de investigación**

- Según el propósito y la naturaleza del problema: **Aplicada**, debido a que la presente investigación busca una alternativa de solución práctica a la problemática planteada en relación a las deficiencias en el entendimiento y la comunicación con personas que presentan limitaciones auditivas o del habla, para ello se considera el uso de la Visión Computacional como herramienta para la traducción del lenguaje de señas a texto en idioma español. (Egg, 2011)

- Según el nivel de conocimientos que se adquieren: **Exploratoria.**  
(Sampieri, Collado, & Lucio, 2010), ya que a través de su estudio se puede dar inicio a nuevas investigaciones donde la visión computacional es empleada como herramienta para el reconocimiento de patrones o la identificación de imágenes en otras áreas de aplicación.

#### ❖ **Método de investigación**

La presente investigación se basa en el método Inductivo – Deductivo, donde se inicia con la inducción de principios explicativos a partir de una realidad observable sobre la traducción en tiempo real del Lenguaje de Señas a texto en idioma español, para luego construir enunciados que contengan y se refieran a la realidad observada y el efecto que tuvo la Visión Computacional en este escenario.

#### ❖ **Delimitación del estudio**

El presente estudio está orientado al uso de la Visión Computacional para la traducción en tiempo real del lenguaje de señas a texto en idioma español, considerando para esta investigación solo la traducción de las letras del alfabeto que son representadas por una sola mano en el Sistema Dactilológico Peruano.

❖ **Régimen del estudio**

Es libre, porque nace de la iniciativa de los investigadores de realizar un estudio de investigación que busque hacer uso de la Visión Computacional para la traducción en tiempo real del lenguaje de señas a texto en idioma español.

❖ **Área de desarrollo del estudio**

Departamento: Ancash

Provincia: Santa

Distrito: Nuevo Chimbote

❖ **Lugar de investigación**

Centro de Educación Básica Especial N° 1 – Zona Norte

Chimbote – Ancash – Perú.

# CAPÍTULO I: LA INSTITUCIÓN

## 1.1. Descripción

### 1.1.1. Razón social

Centro de Educación Básica Especial N° 01

### 1.1.2. RUC

0570226

### 1.1.3. Tipo de Empresa

Institución Educativa del estado

### 1.1.4. Actividad Comercial

Actividades Educativas Especiales

Ugel: UGEL Santa

Nivel: Educación Especial

Turno: Continuo sólo en la mañana

Tipo: Pública de gestión directa

### 1.1.5. Ubicación Geográfica

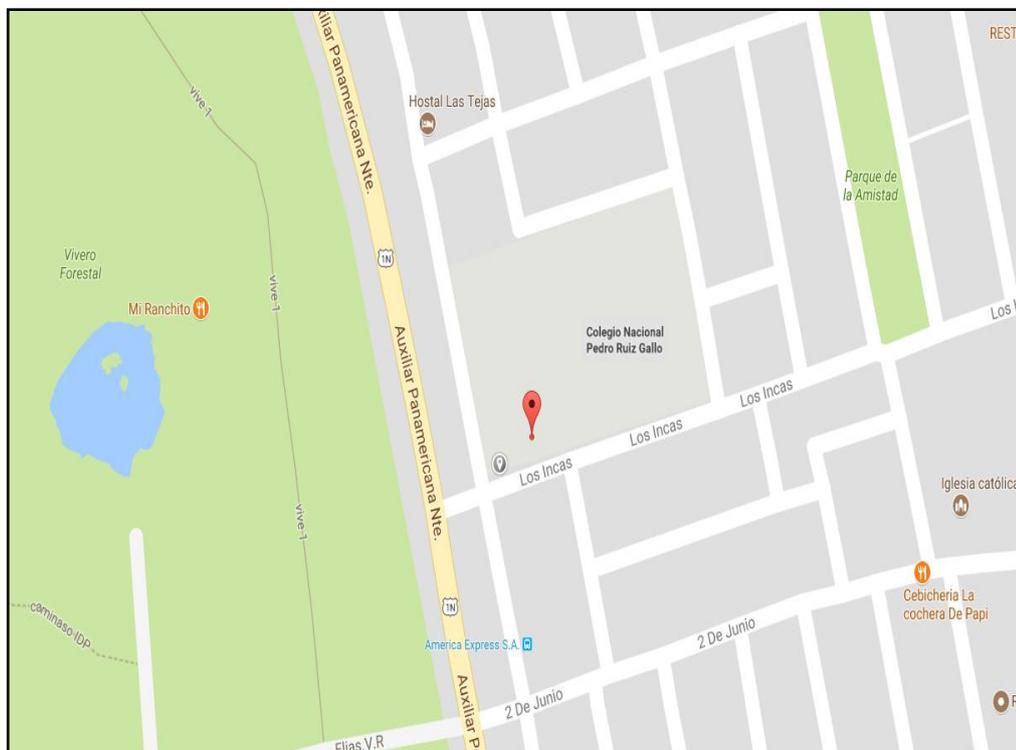
Distrito: Chimbote

Provincia: Santa

Región: Ancash

Ubigeo: 021801

Área: Urbana



*Ilustración N° 1: Ubicación Geográfica CEBE N° 01*

*Fuente: Google Maps*

### 1.1.6. Dirección Legal

MZ P LOTE 24-25 Laderas del Norte - Chimbote

### **1.1.7. Finalidad**

El Centro de Educación Básica Especial N° 01 busca brindar una educación de calidad a niños y jóvenes con necesidades educativas especiales, poniendo en práctica la constante innovación en el desarrollo de los servicios educativos.

Para ello cuenta con docentes capacitados en ofrecer una enseñanza inclusiva, con el fin de que los estudiantes puedan desenvolverse de forma independiente en su entorno social.

### **1.1.8. Estructura Orgánica**

#### **1.1.8.1. Órgano de control**

UGEL Santa

#### **1.1.8.2. Órgano de Dirección**

Director

#### **1.1.8.3. Órgano de Apoyo Administrativo y de Servicio**

- ✓ Apoyo Logístico
- ✓ Secretaria
- ✓ Personal de Servicio

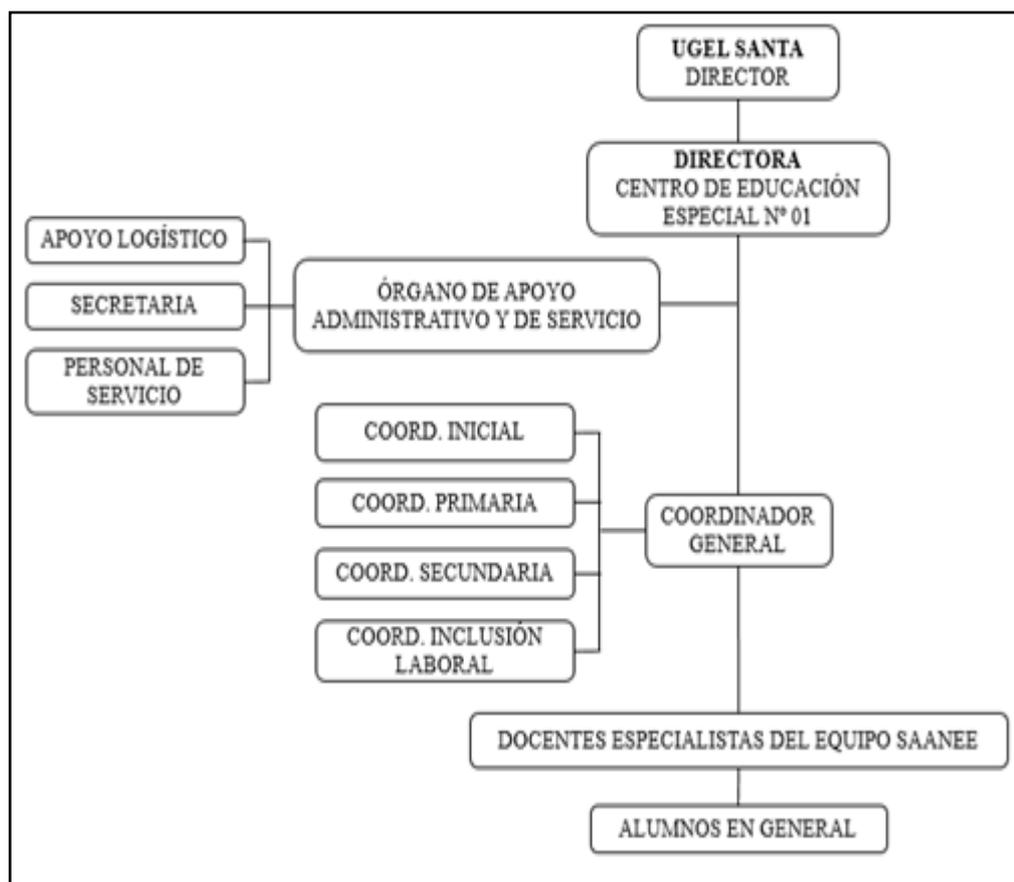
#### **1.1.8.4. Órgano de Línea**

COORDINADOR ACADÉMICO

- ✓ Coordinador del Nivel Inicial

- ✓ Coordinador del Nivel Primaria
- ✓ Coordinador del Nivel Secundaria
- ✓ Coordinador del Nivel Inclusión Laboral

### 1.1.9. Organigrama



*Ilustración N° 2: Organigrama Estructural del CEBE N° 01*

*Fuente: CEBE N° 01*

## 1.2. Direccionamiento Estratégico

### 1.2.1. Visión

El Centro de Educación Básica Especial N° 01 al 2019, será líder en la atención a estudiantes con discapacidad moderada, severa y multidiscapacidad, contará con un Equipo Directivo que ejerza un

liderazgo pedagógico, con docentes capacitados, innovadores, comprometidos en la práctica pedagógica y en la preparación de aprendizajes funcionales, sociales y ocupacionales.

Con un equipamiento adecuado, asumiendo valores como el amor, respeto, igualdad y responsabilidad, aplicando un enfoque inclusivo basado en un modelo social considerando la igualdad de oportunidades, para una inclusión familiar y laboral, enfatizando la conciencia ambiental que busca el desarrollo de capacidades y competencias para el aprovechamiento sostenible de los recursos naturales, realizando un trabajo colaborativo y coordinado con la familia y aliados estratégicos de nuestra localidad.

### **1.2.2. Misión**

El Centro de Educación Básica Especial N° 01 es una I. E. que atiende a estudiantes con necesidades educativas especiales, que practica la innovación permanente en el desarrollo de sus servicios educativos, contando con personal docente con vocación de servicio e identificados con la modalidad y comprometidos en el desarrollo del proceso enseñanza aprendizaje, con un enfoque inclusivo basado en un modelo social, en la formación integral de los estudiantes en la práctica de valores, como personas independientes y autónomas para desarrollarse e integrarse a la familia, escuela y comunidad.

### 1.2.3. Valores

- **Compromiso**

Con los padres de familia que confían en nuestra institución educativa y con los estudiantes a quienes les brindamos una educación especializada e inclusiva, que les permita desarrollarse de forma autónoma en la sociedad.

- **Respeto**

A través de una metodología de enseñanza aprendizaje donde se aplique un enfoque inclusivo, impartiendo el respeto hacia los demás y la igualdad de oportunidades para todos.

- **Innovación**

Crecimiento constante y adaptación a nuevas metodologías de enseñanza, con docentes comprometidos en la práctica pedagógica.

### 1.3. Objetivos de la Institución

#### 1.3.1. Objetivos Estratégicos Institucionales

- Fortalecer en un 90% la cultura de INNOVACIONES PEDAGÓGICAS para la atención a estudiantes severos y multidiscapacitados, que involucren a docentes, padres de familia, demás familiares y alumnos, potenciando y estimulando aprendizajes funcionales para el logro de su autonomía.
- Vivencia en una 90% un CULTURA DE VALORES con toda la comunidad educativa a través de proyectos.

- Fortalecer en un 90% las ESCUELAS DE FAMILIAS a través de talleres a nivel institucional y de aula, involucrando a cuidadores y familiares.
- Continuar en un 95% con la SENSIBILIZACIÓN dirigida hacia la comunidad, padres de familia, autoridades educativas, IER, EBA, ETP., instituciones gubernamentales y no gubernamentales, mediante charlas, testimonios.
- Continuar en un 80% con las campañas de difusión de los servicios que brinda el CEBE y el SAANE, a través de medios de comunicación masiva.
- Realizar campañas de CAPTACIÓN de estudiantes con necesidades educativas especiales, involucrando a todos los profesionales de EBE y SAANE, en un 70%.
- Capacitar en un 90% a los docentes en el uso de la informática educativa y el manejo de programas para optimizar el aprendizaje de los alumnos con necesidades educativas especiales.
- Crear e implementar en un 80% aulas funcionales teniendo en cuenta las características de nuestros estudiantes, donde exploten y desarrollen su potencial, con participación de los padres de familia.
- Adaptar en un 80% la infraestructura existente a las necesidades de nuestros estudiantes, a través de la construcción de rampas, pasamanos, paralelas, etc.

- Continuar en un 90% con las capacitaciones dirigidas a las IEBR, EBA, ETP, EBE, referente al proceso de inclusión escolar y atención a estudiantes con N.E.E.

## **CAPÍTULO II: PLANTEAMIENTO DEL PROBLEMA**

### **2.1. Problema**

#### **2.1.1. Realidad Problemática**

Desde tiempos inmemorables los seres humanos siempre han buscado la forma de poder comunicarse entre ellos, es por eso que han inventado diversas maneras y medios para lograr ese objetivo. Esto convierte a la comunicación en un pilar fundamental para el desarrollo humano.

La comunicación no verbal es empleada especialmente por personas que sufren de limitaciones auditivas o del habla, que, si bien poseen capacidades diferentes, es el lenguaje de señas lo que les permite interactuar con personas de su entorno, siempre y cuando esas personas conozcan y entiendan esta forma de comunicación, sin embargo, la mayor parte de la sociedad no conoce y mucho menos logra entender cuando otra persona hace uso del lenguaje de señas para expresarse.

Según el Observatorio Nacional de la Discapacidad y el INEI en el 2012 existía 532 mil 209 personas con limitaciones auditivas el cual

representaba 1.8% de la población y para el año 2015 se incrementó aproximadamente a 560 mil 730 de un total de 31 151 643 personas. (Observatorio Nacional de la Discapacidad , 2016).

Y a nivel mundial, Según la Organización Mundial de la Salud (OMS), existen 360 millones de personas con algún tipo de discapacidad auditiva, lo cual representa el 5.3% de la población mundial. (Organización Mundial de la Salud , 2012).

Podemos notar que el número de personas con limitaciones auditivas está incrementando a nivel nacional e internacional, si bien es cierto el gobierno peruano viene haciendo esfuerzos para incluir a estas personas a la sociedad, aún sigue existiendo una gran barrera que tenemos que derribar para superar la discriminación social que existe contra estas personas, es con este objetivo que nace la presente investigación, que busca brindar un medio más ágil de comunicación para estas personas y que así puedan contar con más oportunidades de superación.

Para ello se hace uso de las técnicas de visión por computadora con la finalidad de intentar simular la visión humana, de esta manera podemos traducir en tiempo real el lenguaje de señas a texto en idioma español, haciendo uso de librerías basadas en las posiciones de las manos cuando una persona se comunica a través del lenguaje de señas,

es así como las personas que desconocen de este tipo de lenguaje se pueden comunicar con las personas sordomudas de una forma más ágil y efectiva; esto va permitir superar los siguientes aspectos de la problemática en cuestión:

- Exclusión de las personas con limitaciones auditivas o del habla, esto se debe a que no existe una comunicación fluida entre las personas sordomudas y su entorno social, debido a que las personas no logran entender claramente el lenguaje de señas, lo que a su vez impide que las personas sordomudas puedan desenvolverse fácilmente en la sociedad.
- Desconocimiento del lenguaje de señas, puesto que la mayoría de personas no son conscientes de las dificultades que atraviesan las personas sordomudas, no hay un interés general por aprender el lenguaje de señas, esto origina desigualdad en la comunicación.
- Falta de tolerancia a las limitaciones de otra persona, debido a que no existe una concientización de las dificultades que atraviesan día a día las personas que no pueden comunicarse de igual manera en que lo hace una persona que puede oír y hablar sin ninguna dificultad.

### **2.1.2. Análisis del Problema**

Considerando el diagnóstico de la problemática e incidiendo particularmente en un deficiente estado actual de las personas con limitaciones auditivas o del habla, debemos analizar cada una de las

problemáticas identificadas, para profundizar a mayor detalle este problema y posibles soluciones en cuanto a su segmentación:

- Al indicar exclusión de las personas con limitaciones auditivas o del habla, se propone el uso de la visión computacional como herramienta para mejorar la comunicación entre las personas sordomudas y su entorno, a través de la traducción en tiempo real de las principales posiciones de la mano en el lenguaje de señas.
- Considerando el desconocimiento del lenguaje de señas, el desarrollo de esta investigación dará a conocer las técnicas de la visión computacional para aplicarlo en la traducción del lenguaje de señas a texto en idioma español con la finalidad de que cualquier persona pueda interpretar correctamente lo que quiere expresar una persona sordomuda sin necesidad de tener amplios conocimientos sobre este lenguaje.
- Tomando en cuenta la falta de tolerancia a las limitaciones de otra persona, esta investigación busca proponer una herramienta tecnológica que logre una comunicación más ágil y efectiva entre las personas con limitaciones auditivas o del habla y su entorno social, esto con la finalidad de que la comunicación sea fluida y cómoda para ambas partes.

## **2.2. Antecedentes**

- **Local:**

A nivel local ha sido poca o casi nula la investigación en el campo del procesamiento de imágenes y visión computacional, razón por la cual no se ha podido encontrar referencias anteriores.

- **Nacional:**

**a. Detección de Caracteres de Placas de Automóviles Mediante Técnicas de Visión Artificial.**

Autor:

Bach. George Antonio Mundaca Vidarte

Esta investigación presenta el desarrollo de un sistema para la identificación de caracteres en las placas de los automóviles sin importar la perspectiva de la cámara.

El sistema se ha implementado sobre distintas técnicas de procesamiento de imágenes como la variación de gradientes, filtro Sobel, transformada de Bottom-Hat, transformada de Hough, transformada proyectiva bidimensional y finalmente realiza el reconocimiento de los caracteres usando el método de correlación de Pearson; sin embargo, sus resultados dependen de una evaluación con escenarios estáticos y una distancia prudente. Por último, este proyecto obtiene un margen de error del 23% que según la investigación es aceptable (Mundaca Vidarte, 2016).

Este antecedente brinda información detallada de las principales técnicas de procesamiento de imágenes, lo cual muestra un

panorama más amplio de las diferentes aplicaciones de la visión computacional y permite discernir sobre las técnicas que son más adecuadas para el proyecto.

**b. Caracterización y Reconocimiento de Objetos Mediante Algoritmos de Visión Computacional para la Interacción de un Robot con su Entorno.**

Autor:

Dr. César Armando Beltrán Castañón

Este proyecto muestra un enfoque usando dos métodos de procesamiento de imágenes bien conocidos que son SIFT (Scale-Invariant Feature Transform) y SURF (Speeded-Up Robust Features) los cuales tiene similitud en los pasos que siguen para obtener la información de un objeto.

Este proyecto se centra en reconocer varios objetos a la vez a partir de una imagen, y esto es posible detectando los puntos de interés a partir de los máximos y mínimos de la Diferencia de Gaussianos, cabe resaltar que las pruebas con el prototipo se dieron en un ambiente controlado libre de colores en fondo.

Lo destacable en este proyecto es que para el reconocimiento de los objetos se basó en dos partes (ya que utilizaron video y no fotografía digital), la primera detección de la información de un objeto en la escena y la segunda predicción del clasificador entrenado con un dominio de objetos (Robles Pizarro, 2016).

Lo que podemos rescatar del presente antecedente para nuestro proyecto es la forma de cómo trabajar con videos para capturar y procesar un fotograma de esta manera dar respuesta inmediatamente a la seña que se proyecta en el video (traducir en tiempo real) y no con fotografías digitales ya tomadas.

- **Internacional:**

**a. Diseño de un Sistema de Reconocimiento de Gestos no Móviles Mediante el Procesamiento Digital de Imágenes (Universidad Autónoma del Caribe).**

Autores:

Bach. Valeria Valencia Sierra

Bach. Betsy Villa Brochero

Este proyecto fue desarrollado en Barranquilla-Colombia, tiene como fin procesar imágenes ya capturadas (fotografías), de las posiciones del lenguaje de señas colombiano, para luego compararlo con su base datos de fotográficas y poder arrojar un resultado de la comparación la cual es la letra que representa esa imagen, su presión de acierto es de 77% dejando un margen de error de 23%.

Los autores buscaron y estudiaron la metodología adecuada para poder desarrollar de la mejor manera los procesos que intervienen en el trabajo de procesamiento de imágenes, con el objetivo de reducir el margen de error (Valencia Sierra & Villa Brochero, 2014).

Este antecedente brinda la forma y la metodología con la podemos trabajar el procesamiento de imágenes y la visión computacional además está orientado al mismo fin del presente proyecto la cual es traducir el lenguaje de señas a texto, con la única diferencia que en nuestro proyecto se plantea la traducción en tiempo real (ya no de fotografías y sino de video).

**b. Reconocimiento automático de lenguaje de signos: Lenguaje ASL (Universidad de Barcelona).**

Autor: Enrique Antón López

Este trabajo está más orientado en el estudio de los diferentes algoritmos que se puedan utilizar para tener más precisión y velocidad a la hora que procesar las imágenes.

En tal sentido el autor experimento con varios algoritmos pero el siendo más óptimo el DTW (Dinamic Time Warping), el cual permite una mejor segmentación de regiones de interés y proyección de contornos para la extracción de características de

la imagen a procesar, por último en sus conclusiones especifica que si el reconocimiento se hace en un ambiente no controlado (en escenarios diversos por ejemplo con bastante o poca luminosidad, diferentes fondos etc.) la probabilidad de acertar con la letra correcta disminuiría (Antón López, 2009).

De este antecedente se puede rescatar los diferentes algoritmos que se aplican para tener un reconocimiento más limpio y rápido, además indica que debe existir un mayor enfoque en el proceso de segmentación por umbral ya que es la clave para un reconocimiento preciso, y por último tomar muy en serio en qué ambiente se desarrollara los experimentos con el prototipo.

### **2.3. Alcance**

Este proyecto se centrará en la traducción de las letras del alfabeto que son representadas por una sola mano en el lenguaje de señas peruano, a texto en idioma español, considerando principalmente el reconocimiento de las letras del abecedario, ya que representa la base de todo sistema de comunicación. Sin embargo, esto no significa que los resultados de esta investigación no puedan ser útiles para investigaciones futuras donde se busque la traducción de posiciones más complejas de las manos en el lenguaje de señas.

## **2.4. Importancia de la investigación**

Establece las bases para la traducción del lenguaje de señas a través de la Visión Computacional, abriendo nuevos caminos para futuras investigaciones donde esta herramienta tecnológica solucione situaciones más complejas del proceso de traducción y mejore ampliamente la comunicación entre las personas con limitaciones auditivas o del habla, y su entorno, permitiéndoles una mayor inclusión y un mejor desenvolvimiento en la sociedad.

## **2.5. Formulación del problema**

¿En qué medida la visión computacional permitirá la traducción en tiempo real del lenguaje de señas a texto en idioma español?

## **2.6. Formulación de hipótesis**

### **2.6.1. Hipótesis**

La visión computacional permite traducir en tiempo real el lenguaje de señas a texto en idioma español.

### **2.6.2. Operacionalización de las variables**

**Variable Independiente (VI):** Visión computacional.

**Variable Dependiente (VD):** Traducción en tiempo real del lenguaje de señas a texto en idioma español.

Tabla N° 1: Indicadores de las variables del proyecto.

<b>Variables</b>	<b>Indicadores</b>	<b>Unidad de Medida</b>
<b>Variable Independiente</b>	➤ Nivel de impacto.	➤ Porcentaje.
<b>Variable Dependiente</b>	➤ Nivel de rapidez de la traducción. ➤ Nivel presión de la traducción.	➤ Segundos. ➤ Porcentaje.

## 2.7. Objetivos de la investigación

### 2.7.1. Objetivo General

Traducir en tiempo real las principales posiciones de la mano en el lenguaje de señas a texto en idioma español.

### 2.7.2. Objetivo Específico

- Obtener información pertinente del lenguaje de señas peruano, para el desarrollo del algoritmo de traducción.
- Verificar la efectividad de los algoritmos para reconocimiento de imágenes mediante pruebas con el modelo de Visión Computacional.
- Elaborar una base de datos de las posiciones de la mano que representan las letras del alfabeto dactilológico peruano, con las imágenes recolectadas en el proceso de investigación.
- Optimizar la precisión en la traducción del lenguaje de señas a texto en idioma español con el uso de la Visión Computacional.

- Mejorar el tiempo empleado en la traducción de una determinada letra del lenguaje de señas a través del modelo de Visión Computacional
- Verificar la usabilidad del modelo de Visión Computacional como herramienta tecnológica para la traducción del lenguaje de señas.

## **CAPÍTULO III: MARCO TEÓRICO**

### **3.1. Marco Conceptual**

#### **3.1.1. Visión Computacional o Artificial**

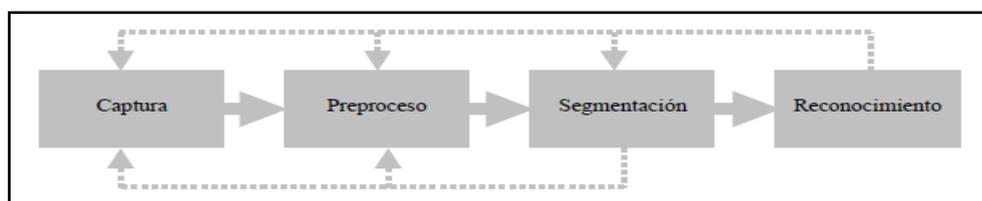
La visión artificial se basa en el reconocimiento automático de las propiedades y estructura de un contexto tridimensional dinámico, a partir de una o varias imágenes bidimensionales del contexto representado. En este proceso se unen diferentes conceptos de la física como el color, óptica, electrónica, geometría, algorítmica, sistemas de computación, etc. donde el estudio exhaustivo de cada uno de los conceptos es determinante para un buen trabajo. (Alegre Gutiérrez, Pajares Martinsanz, & De la Escalera Hueso, 2016).

##### **3.1.1.1. Etapas de un sistema de visión artificial:**

La visión computacional (o visión artificial), intenta reproducir el comportamiento de la visión humana por tal

razón se define tradicionalmente (Tratando de imitar la visión humana) en cuatro fases principales:

- ✓ Primera fase; es donde, mediante algún tipo de sensor, se realiza la captura o adquisición de las imágenes digitales.
- ✓ Segunda fase; con el objetivo de facilitar las etapas posteriores se procede con el tratamiento digital de las imágenes. Este procesamiento previo, elimina partes indeseables de la imagen o realza partes interesantes de la misma mediante filtros y transformaciones geométricas.
- ✓ Tercera fase; también llamada segmentación, se basa en el aislamiento de los elementos que interesan de una escena para comprenderla.
- ✓ Cuarta fase; consiste en el reconocimiento o clasificación. Es donde se diferencian los objetos segmentados, basados en las características previamente establecidas para su distinguirlos.



*Ilustración N° 3: Etapas de un sistema de visión artificial*

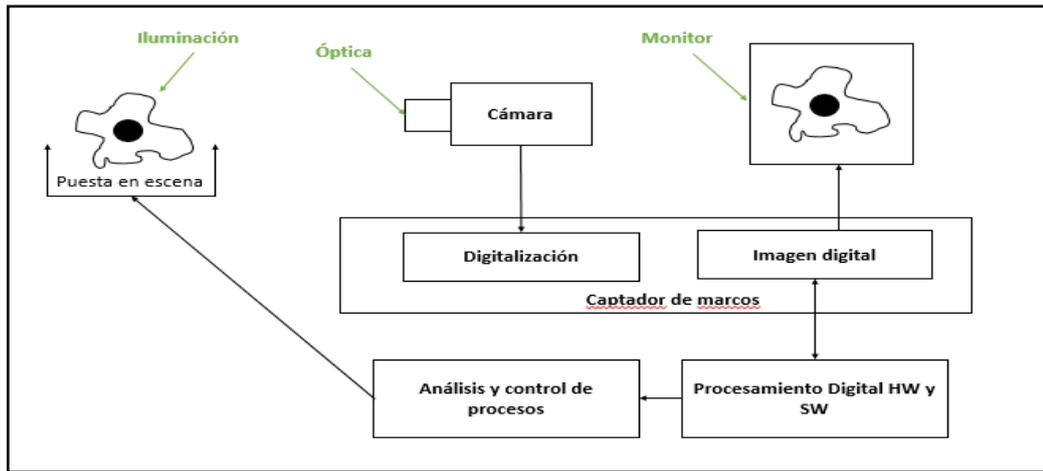
En ocasiones estas cuatro fases deben realimentarse hacia atrás, por lo que no siempre se siguen de manera secuencial.

(Vélez Serrano, Moreno Díaz, Sánchez Calle, & Sánchez-Marín, 2015).

### 3.1.1.2. Componentes de un Sistema de Visión Computacional

- ✓ **Sensor óptico:** El sensor puede ser una cámara monocromo o color que produce una imagen completa del dominio del problema.
- ✓ **Tarjeta de adquisición de imagen:** Digitaliza la señal de video entregada por el sensor óptico, para luego almacenarla y procesarla por la computadora.
- ✓ **Monitor de vídeo:** Permite visualizar tanto las imágenes o escenas captadas como los resultados del procesamiento de dichas imágenes.

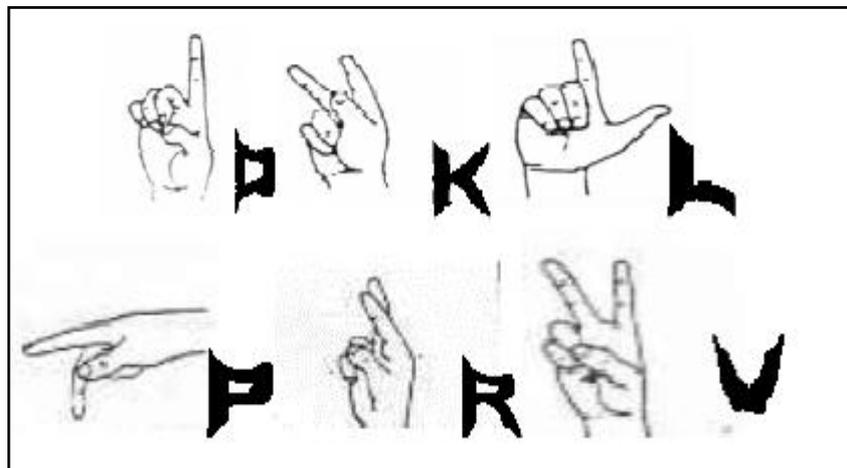
En la figura se puede ver el objeto observado, la iluminación, la cámara y el proceso de digitalización que permite la visualización de la imagen digital, el proceso de extracción de parámetros y su posterior proceso de análisis (González Marcos, y otros, 2006).



*Ilustración N° 4: Diagrama de bloques de un Sistema de Visión Computacional*

### 3.1.2. Lenguaje de Señas

Es un sistema de comunicación de grupos humanos de población sorda en el mundo entero; compuesta por códigos desarrollados de manera visual y gestual. En la comunicación por medio del lenguaje de señas, participan básicamente las manos, el rostro y la parte superior del cuerpo humano, debido a la imposibilidad que tienen las personas sordas para utilizar del oído y la voz como canales de comunicación (OVIEDO, 1998).



*Ilustración N° 5: Lenguaje de señas*

### 3.1.2.1. Dactilología

Es la representación gestual de las letras del alfabeto, empleada sobre todo para el deletreo de ciudades, conceptos nuevos, direcciones, palabras que no tienen signo asignado y nombres propios cuando se habla de una persona desconocida.

Generalmente, la dactilología es empleada como sistema alternativo a otros sistemas de comunicación y no como principal, ya que requiere un gran esfuerzo atencional y su ejecución es lenta.



*Ilustración N° 6: Sistema dactilológico peruano (español)*

**Fuente:** (Ministerio de Educación, Peruano, 2010)

## 3.2. Marco Teórico

### 3.2.1. Procesamiento digital de imágenes

Es una rama de la electrónica que tiene dos aplicaciones principales: mejorar la información descrita en una imagen para la percepción humana y procesar la imagen para su almacenamiento, transmisión y representación para la percepción de máquinas autónomas (González Marcos, y otros, 2006).

#### 3.2.1.1. Formación y Representación de la Imagen

Una imagen no es más que un objeto iluminado por una fuente radiante. La naturaleza de una imagen es representada por  $f(x,y)^2$ , está caracterizada por dos componentes que reciben el nombre de iluminación y reflectancia, notándose  $i(x,y)$  y  $r(x,y)$  respectivamente. Ambas funciones se combinan como producto para dar  $f(x,y)$ .

Los elementos que intervienen en la formación de la imagen son: el objeto, la fuente radiante y el sistema de formación de la imagen (sensor óptico). La imagen digital puede ser representada por una matriz  $f$  de dimensiones  $N \times M$  de la forma:

$$f = \begin{bmatrix} f(1,1) & f(1,2) & \dots & f(1,M) \\ f(2,1) & f(2,1) & \dots & f(2,M) \\ \vdots & \vdots & \dots & \vdots \\ f(N,1) & f(N,2) & \dots & f(N,M) \end{bmatrix}$$

Donde cada elemento (pixel) da la intensidad de la imagen en ese punto. (González Marcos, y otros, 2006).

### 3.2.1.2. Vecinos de un Pixel:

Un pixel  $p$  de coordenadas  $(x, y)$  tiene cuatro vecinos horizontales y verticales cuyas coordenadas son:

$$(x + 1, y), (x - 1, y), (x, y + 1), (x, y - 1)$$

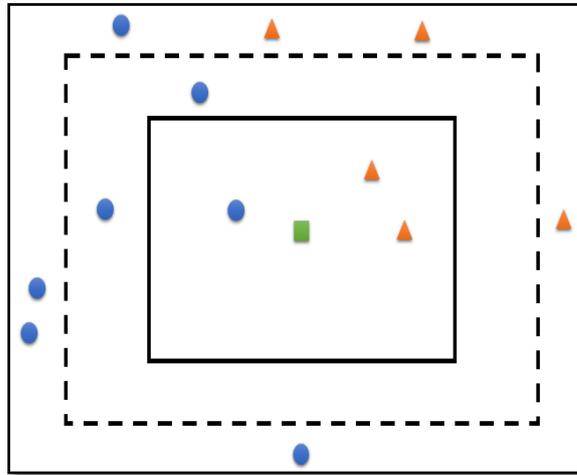
Este conjunto de pixeles, que recibe el nombre de 4-vecinos de  $p$ , se nota  $N_4(p)$ . Cada pixel está a distancia unitaria de  $(x, y)$ , y algunos de los vecinos de  $p$  caen fuera de la imagen digital si  $(x, y)$  está en el borde de la imagen. Los pixeles diagonales vecinos de  $p$  tienen coordenadas:

$$(x + 1, y + 1), (x - 1, y - 1), (x - 1, y + 1), (x + 1, y - 1)$$

y se notan  $N_D(p)$ . Estos puntos, junto con los cuatro vecinos, se llaman 8-vecinos de  $p$  y se notan  $N_8(p)$ . (González Marcos, y otros, 2006).

Es decir, es como un sistema de elecciones donde se toma un número  $k$  de las muestras más cercanas, de estas  $k$  muestras cercanas se cuentan las de mayor número que

pertenezcan a una misma clase y basado a esto se asigna la clase.



*Ilustración N° 7: Ejemplo del Algoritmo K-vecino más cercano*

Podemos observar que el cuadrado menor tiene  $k = 3$ , por tanto, es clasificado con los triángulos ya que estos tienen el mayor número de muestras ( $m=2$ ); en cambio para el cuadro mayor tiene  $k= 5$ , por ende, es clasificado con los círculos, porque corresponde al mayor número de muestras ( $m=3$ ).

### **3.2.2. Filtro Digital**

Puede ser representada como una operación matemática que toma una secuencia de números (la señal de entrada) y la altera produciendo otra secuencia de números (la señal de salida) con el fin de resaltar o atenuar ciertas características (Steven W, 1997).

#### **3.2.2.1. Filtrado en el dominio de la frecuencia**

Un ejemplo es la de transformada de Fourier de una función, la cual representa el espectro de frecuencias de la misma. Por lo tanto, a cambios bruscos produce componentes de alta frecuencia, de lo contrario sus componentes tienen baja frecuencia.

En el ámbito del tratamiento de imágenes, la función  $f(x, y)$  representa los niveles de intensidad de los píxeles a tratar y se define la transformada discreta de Fourier en 2D como se muestra en la ecuación (C. Gonzalez & E. Woods, 2002).

$$F(u, v) = \frac{1}{MN} \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} f(x, y) e^{-j2\pi(\frac{ux}{N} + \frac{vy}{M})}$$

✓ **Filtros paso bajo**

Es aquel que deja pasar las bajas frecuencias y atenúa o eliminan las altas frecuencias. Su efecto en el dominio del espacio es un suavizado de la imagen, es decir, la eliminación de ruido y otros pequeños detalles en la imagen. Un ejemplo es el filtro paso bajo Gaussiano. (Steven W, 1997).

✓ **Filtros paso alto**

Es lo contrario que el paso bajo, dejan pasar las altas frecuencias y atenúan o anulan las bajas frecuencias,

provocando resaltes en las zonas con cambios bruscos en los niveles de intensidad. (Steven W, 1997).

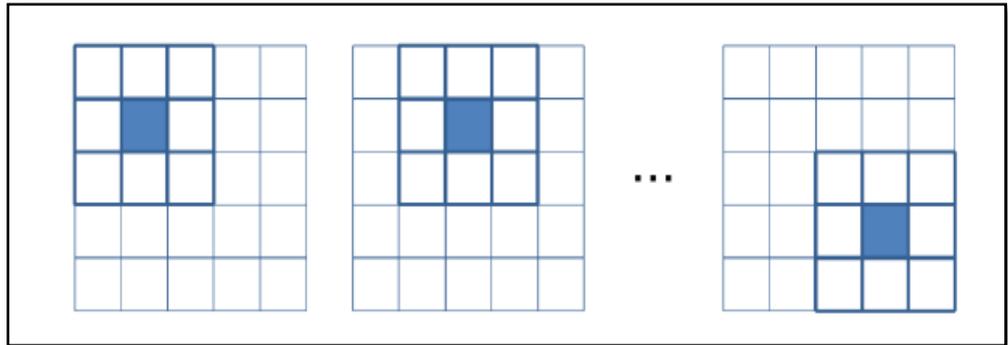
✓ **Filtros paso banda**

Los filtros paso banda son aquellos que atenúan o eliminan las bajas y altas frecuencia dejando pasar aquellas que pertenecen a una banda determinada (Steven W, 1997).

**3.2.2.2. Filtrado en el dominio del espacio**

En este caso las operaciones son aplicadas directamente sobre los píxeles de la imagen y no sobre su transformada en el dominio frecuencial. Para obtener el nuevo valor de cada uno de los píxeles se utiliza el píxel referencia y una serie de píxeles localizados dentro de una ventana alrededor de éste.

Los filtros lineales están basados en los llamados kernels. Un kernel no es más que una matriz de coeficientes que asigna una serie de valores a los píxeles vecinos del píxel de referencia  $I(i, j)$ . La cual funciona como una ventana móvil que recorre toda toda la imagen (Acharya & K. Ray, 2005).

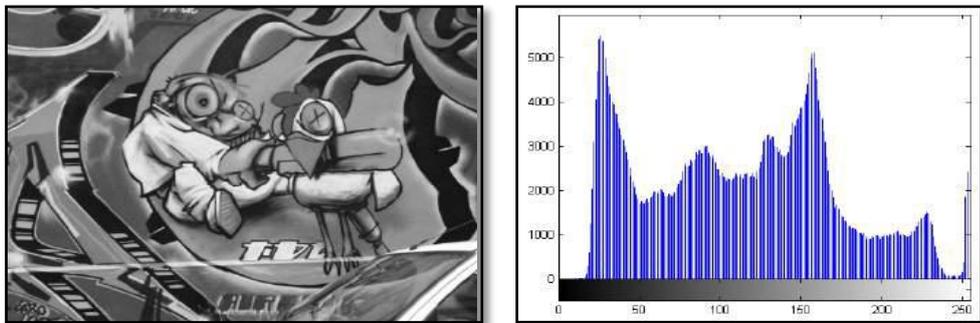


*Ilustración N° 8: Representación de un kernel y el movimiento que realiza para recorrer toda la imagen.*

También existen los tipos de filtrado de paso bajo y alto (Teniendo la misma conceptualización que la del filtrado en el dominio de frecuencia).

### 3.2.3. El histograma de la imagen

El histograma de una imagen es la representación de la frecuencia de aparición de cada uno de los valores. Sabiendo que el 0 representa el negro y el 255 el blanco (En una imagen de escala de grises), el histograma muestra desde los píxeles oscuros a los claros (Alegre Gutiérrez, Pajares Martinsanz, & De la Escalera Hueso, 2016).



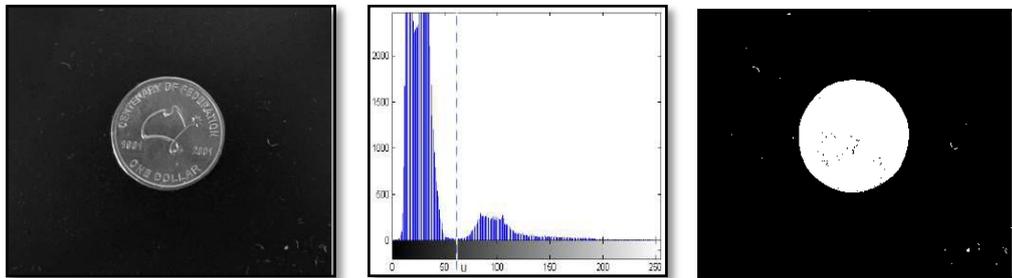
*Ilustración N° 9: Imagen junto a su correspondiente histograma.*

### 3.2.3.1. Umbralización del histograma

El resultado este proceso es una imagen binarizada. Sean  $I$  e  $I'$  la imagen original y su imagen umbralizada,  $U$  el valor de intensidad establecido como umbral y  $L - 1$  el mayor nivel de intensidad posible, cada nuevo píxel  $I'_{ij}$  se obtiene mediante:

$$I'(i, j) = \begin{cases} 0, & I(i, j) < U \\ L - 1, & I(i, j) \geq U \end{cases}$$

Se puede observar en la figura de una moneda en la que, al aplicar un umbralizado binario con  $U=60$ , es posible separar la moneda del fondo.

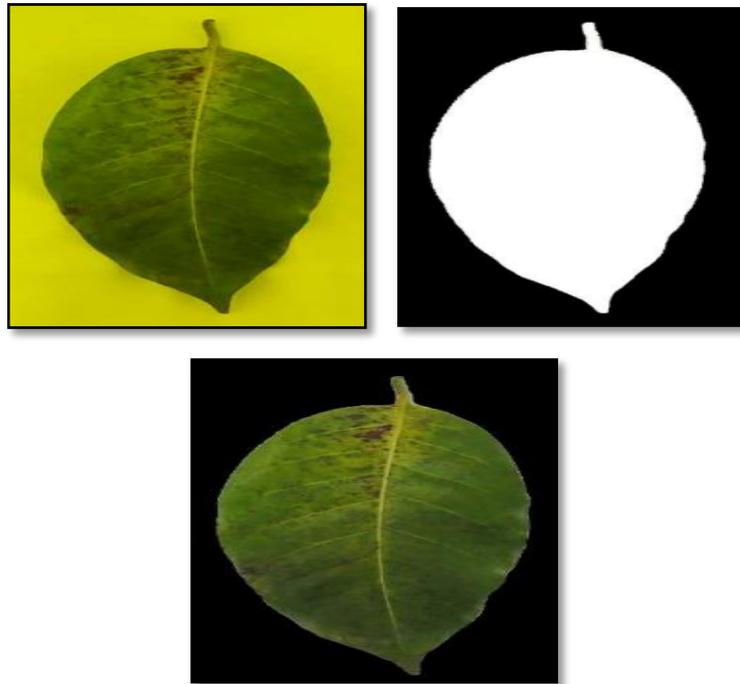


*Ilustración N° 10: Imagen junto al resultado de su umbralización con  $U=60$ .*

### 3.2.4. Segmentación

El fin de esta etapa es separar los objetos de interés de una la imagen digital, es decir, consiste en diferenciar los diversos objetos y dónde se encuentran del fondo de la imagen, la etapa de segmentación es crucial para el reconocimiento de formas pudiéndose complicar o

simplificar enormemente según sea la escena más o menos compleja (González Marcos, y otros, 2006).



*Ilustración N° 11: Proceso de segmentación automática de la imagen*

▪ **Dilatación Binaria**

Dados dos conjuntos  $A$  y  $B$  de  $Z^2$ , la dilatación, denotada por  $A \oplus B$ , se define como:

$$A \oplus B = \{ c \in E^N \mid c = a + b \text{ para todo } a \in A \text{ y } b \in B$$

Dada una máscara  $B$ , la dilatación de  $A$  por  $B$  es el conjunto de todos los desplazamientos de  $x$  tales que  $B$  y  $A$  se solapan en al menos un elemento distinto de cero. Por ejemplo:

0	1	0
1	1	1
0	1	0

*Ilustración N° 12: Dilatación binaria Máscara B.*

Y una figura A formada por una matriz:

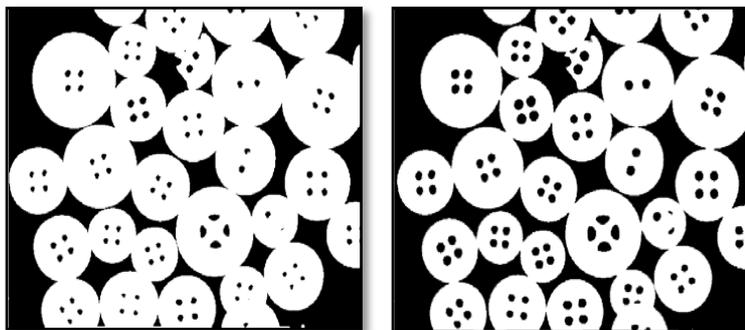
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	1	1	1	0	0
0	0	1	0	1	0	0
0	0	1	1	1	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

*Ilustración N° 13: Dilatación binaria Imagen A*

En otras palabras, la dilatación consiste en ir desplazando el elemento estructurante ( $B$ ) por toda la imagen, y si alguno de los elementos de la matriz (denominados vecinos de la máscara) coincide con un pixel del entorno, entonces el pixel donde está centrado en  $A$  se pone a uno.

0	0	0	0	0	0	0
0	0	1	1	1	0	0
0	1	1	1	1	1	0
0	1	1	1	1	1	0
0	1	1	1	1	1	0
0	0	1	1	1	0	0
0	0	0	0	0	0	0

*Ilustración N° 14: Imagen resultante  $A \oplus B$*



*Ilustración N° 15: Imagen binarizada y dilatación de la misma (derecha).*

▪ **Erosión Binaria**

Dados dos conjuntos  $A$  y  $B$  de  $Z^2$  la erosión, denotada por  $A \ominus B$ , se define como:

$$A \ominus B = \{ c \in E^N \mid x = x + b \in A \text{ para todo } b \in B$$

En otras palabras, la erosión establece a cero todos los pixeles de la imagen que no contengan completamente al elemento estructurante en su entorno.

Si la dilatación expandía los bordes y contornos de los objetos, la erosión reduce los contornos de los objetos. Una de sus aplicaciones es para separar objetos que están unidos por una pequeña parte de sus contornos.

Por ejemplo, usando la máscara del caso anterior:

0	1	0
1	1	1
0	1	0

*Ilustración N° 16: Erosión binaria Máscara B*

Y una figura  $A$  formada por una matriz:

0	0	0	0	0	0	0
0	1	1	1	1	1	0
0	1	1	1	1	1	0
0	1	1	1	1	1	0
0	1	1	1	1	1	0

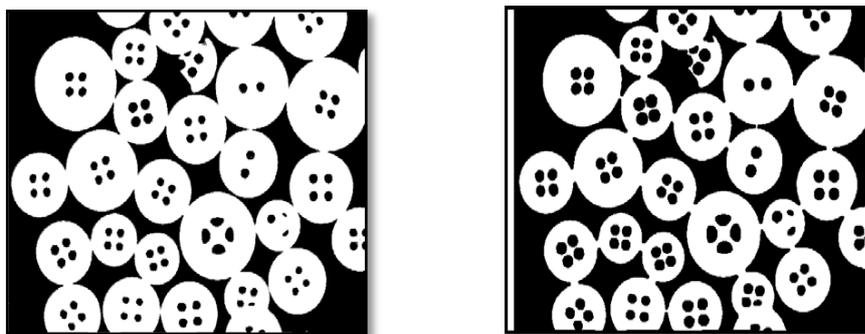
0	1	1	1	1	1	0
0	0	0	0	0	0	0

*Ilustración N° 17: Erosión binaria Imagen A*

El proceso de erosión  $A \ominus B$  consiste, para cada punto  $A(i, j)$ , en solapar la máscara  $B$  en  $A$  (siendo el punto central de  $B$  el  $(i, j)$ ), y ver si todos los puntos a uno de la máscara  $B$  coinciden con los de la imagen, poniéndolo a uno. En caso contrario se pone a cero.

0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	1	1	1	0	0
0	0	1	1	1	0	0
0	0	1	1	1	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

*Ilustración N° 18: Imagen resultante  $A \ominus B$*



*Ilustración N° 19: Imagen binarizada (izquierda) y erosionada (derecha)*

### 3.2.4.1. Técnicas de Segmentación

- ✓ Segmentación Basada en Umbralización

Comprende el proceso de umbralización basado en el histograma que ya se vio anteriormente. Si se analizamos una imagen de una manzana roja en un fondo blanco, el histograma estará formado por dos montañas correspondientes a los píxeles del nivel de gris (imagen a escala de grises) y al fondo blanco. (González Marcos, y otros, 2006).

✓ **Segmentación Basadas en Bordes**

Está basado en los bordes hallados en la imagen por los detectores. Para ello como requisito indispensable se debe conocer los métodos de extracción de bordes. (González Marcos, y otros, 2006).

✓ **Segmentación Orientada a Regiones**

Los métodos para este tipo de segmentación construyen las regiones directamente y los anteriores métodos de las segmentaciones solo encontraban bordes entre regiones. Las regiones finales de estos procesos de segmentación tienen que ser homogéneas y maximales, donde maximal significa que los criterios de homogeneidad no se cumplirán tras la unión de una región con otra. (González Marcos, y otros, 2006).

✓ **Técnicas de Clusterizado**

Esta técnica trata de definir un conjunto de características asociadas a cada píxel creando un vector multidimensional por cada punto de la pantalla. La técnica de clusterizado básicamente se encarga de agrupar dichos vectores según su grado de parentesco entre ellos y así, finalmente se obtienen N grupos de píxeles con características semejantes entre ellos (González Marcos, y otros, 2006).

#### ✓ **Segmentación por Moción**

Este tipo de segmentación se viene estudiando mucho actualmente ya que permite analizar objetos en movimiento. En otras palabras la segmentación se aplica a varias imágenes obtenidas en el tiempo, de manera que por comparación se identifica el objeto en movimiento y se extraen sus características que se desean analizar y almacenar (González Marcos, y otros, 2006).

## CAPÍTULO IV: METODOLOGÍA DE DESARROLLO

### 4.1. Aplicación de la Metodología SCRUM

#### 4.1.1. Fundamentación y Análisis

Si bien se analizó diferentes metodologías, en las que cada una de ellas ofrecía un aporte valioso al desarrollo de la investigación, se escogió la metodología Scrum por ser la que más se adaptaba a las características del proyecto, además de su fácil implementación y la calidad de sus resultados en corto tiempo. Las razones que determinaron el uso de Scrum como metodología ágil de desarrollo son las siguientes:

- **Desarrollo rápido**, ya que permite priorizar los puntos que representan un valor importante para investigación, realizando entregables en periodos cortos de tiempo y optimizando tiempo, recursos y trabajo.
- **Fácil escalabilidad**, con esta metodología el equipo de desarrollo puede mejorar las características de la investigación de una forma ordenada, transparente y sin mucha dificultad.
- **Aplicable a pequeños y medianos proyectos**, donde la asignación tanto de roles como de tareas y las reuniones constantes, ya sea para un equipo de desarrollo pequeño, permiten que la aplicación de la metodología en una investigación relativamente mediana, obtenga resultados óptimos y confiables.

#### 4.1.2. Valores de Trabajo

En Scrum, para lograr un clima laboral adecuado donde el equipo de desarrollo trabaje de forma unida y colaborándose unos a otros con el fin de entregar un producto de calidad, se deben establecer valores que cada miembro del proyecto debe poner en práctica.

Entre los principales valores están el respeto, muy importante para establecer reglas claras en el trabajo, el compromiso más que con el proyecto con el equipo mismo, donde cada uno dé lo mejor de sí para cumplir con el objetivo, y el enfoque preciso en lo que se quiere lograr, priorizando las actividades más relevantes para el negocio y cumpliendo con las entregas en tiempos cortos.

#### 4.1.3. Personas y Roles de Proyecto

- **Product Owner**

Mg. Angélica Margarita Báez López.

- **SCRUM Master**

Leticia Teresa López Olivos.

- **Team**

- **Product Manager** : Leticia Teresa López Olivos.
- **Designer** : Eder Alberto Albino Huertas
- **Developer** : Eder Alberto Albino Huertas
- **Tester** : Leticia Teresa López Olivos

#### **4.1.4. Elaboración del Backlog**

El backlog es el listado de los requerimientos de la investigación y ha sido elaborado por el Product Owner, en colaboración con el Scrum Master, logrando definir los siguientes requerimientos:

- Abstracción y recopilación de datos del lenguaje de señas.
- Definición de las letras del abecedario.
- Toma de fotos de las distintas posiciones de la mano.
- Elección de los algoritmos.
- Elección del lenguaje de programación
- Procesamiento de las imágenes.
- Entrenamiento del modelo de traducción.
- Desarrollo de la interfaz del modelo de traducción.
- Desarrollo del Seleccionador del lenguaje de señas.

#### **4.1.5. Análisis de Complejidad**

Scrum busca resultados en periodos cortos de tiempo, para ello evalúa la complejidad de los ítems del backlog que han sido determinados como viables y prioritarios en el punto anterior, de este modo establecen los tiempos para los requerimientos más resaltantes. La complejidad se mide en base a los recursos y el tiempo que conlleve el desarrollo de un ítem, definiéndose tres niveles:

- Alta: Mayor cantidad de recursos en un tiempo considerable
- Media: Menos recursos en un tiempo moderado.

- Baja: Consumo mínimo de recursos en un corto tiempo

*Tabla N° 2: Complejidad de Requerimientos*

N°	Requerimiento	Complejidad
1	Abstracción y recopilación de datos del lenguaje de señas.	Media
2	Definición de las letras del abecedario.	Baja
3	Toma de fotos de las distintas posiciones de la mano.	Media
4	Elección de los algoritmos.	Media
5	Elección del lenguaje de programación	Media
6	Procesamiento de las imágenes.	Alta
7	Entrenamiento del modelo de traducción.	Alta
8	Desarrollo de la interfaz del modelo de traducción.	Baja
9	Desarrollo del Seleccionador del lenguaje de señas.	Media

#### **4.1.6. Asignación de Valores de Story Points**

Un Story Point es la medida de esfuerzo a través de la cual, el equipo evalúa la complejidad, el nivel de incertidumbre y el riesgo que existe al desarrollar una historia, es decir un requerimiento, basándose en las habilidades y el conocimiento que poseen los miembros del equipo. Cabe resaltar que esta puntuación no está basada en las horas que va a necesitar el requerimiento para ser desarrollado, se basa en la complejidad, la incertidumbre y la habilidad de los miembros del equipo para resolver el problema planteado.

Para este caso, se ha optado como fórmula matemática para determinar las puntuaciones, la secuencia de Fibonacci, ya que el nivel de incertidumbre estará dado por la misma escala, lo que evita sobre-estimación del problema.



*Ilustración N° 20: Gráfico de valores de Story Points*

#### 4.1.7. Valoración de User Story

Los User Story son los requerimientos que han sido definidos en los puntos anteriores, cada User Story va a recibir una puntuación en base a la escala establecida para los Story Points, además se definen los User Story no funcionales del proyecto.

- **User Story Funcionales:**

*Tabla N° 3: Story Points para cada User Story Funcional.*

N°	USER STORY	STORY POINTS
1	Abstracción y recopilación de datos del lenguaje de señas.	13
2	Definición de las letras del abecedario.	5
3	Toma de fotos de las distintas posiciones de la mano.	13
4	Elección de los algoritmos.	21
5	Elección del lenguaje de programación.	13
6	Procesamiento de las imágenes.	55
7	Entrenamiento del modelo de traducción.	34
8	Desarrollo de la interfaz del modelo de traducción.	8
9	Desarrollo del Seleccionador del lenguaje de señas.	21

- **User Story No Funcionales:**

*Tabla N° 4: Story Points para cada User Story No Funcional.*

N°	USER STORY	STORY POINTS
1	Interface amigable y de fácil entendimiento	5
2	Fácil reconocimiento de las distintas posiciones de la mano.	21
3	Exactitud y rapidez en la traducción del lenguaje de señas.	34

#### **4.1.8. User Story con Puntuación Mayor a 20**

Con la finalidad de priorizar los requerimientos, optimizando los esfuerzos del Team Scrum para el rápido desarrollo de los puntos más importantes para el sistema, se realizan filtros de las dos tablas anteriores y se muestran los requerimientos funcionales y no funcionales que obtuvieron un Story Point mayor a 20.

- **User Story Funcionales**

*Tabla N° 5: User Story Funcionales con puntuación mayor a 20*

<b>N°</b>	<b>USER STORY</b>	<b>STORY POINTS</b>
1	Elección de los algoritmos.	21
2	Procesamiento de las imágenes.	55
3	Entrenamiento del modelo de traducción.	34
4	Desarrollo del Seleccionador del lenguaje de señas.	21

- **User Story No Funcionales:**

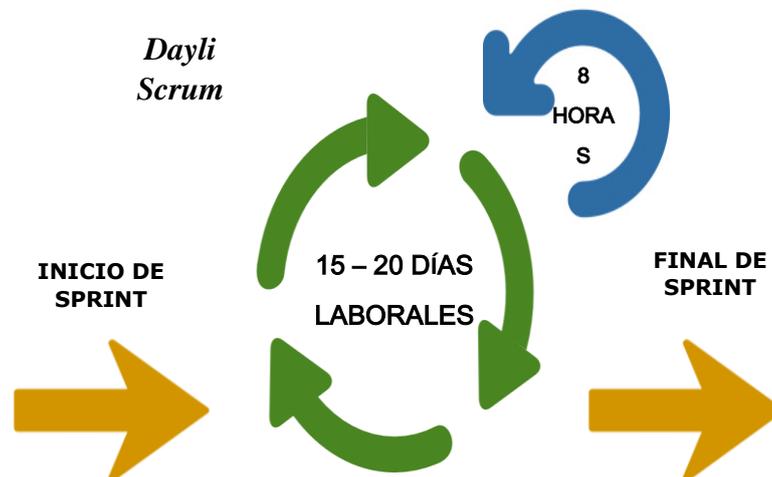
*Tabla N° 6: User Story No Funcionales con puntuación mayor a 20*

<b>N°</b>	<b>USER STORY</b>	<b>STORY POINTS</b>
1	Fácil reconocimiento de las distintas posiciones de la mano.	21
2	Exactitud y rapidez en la traducción del lenguaje de señas.	34

#### **4.1.9. Duración del Sprint**

Con el objetivo de establecer los tiempos para los entregables, se debe definir claramente el intervalo de duración del sprint, teniendo en cuenta el número de integrantes del equipo y sabiendo además que el tiempo establecido para un entregable no debe superar las 4 semanas.

Cada día laborable se realizarán sincronizaciones con todo el equipo para comprobar el progreso de la investigación y conocer los problemas que se presentaron en el desarrollo de los User Story.



*Ilustración N° 21: Gráfico de Duración del Sprint.*

Finalmente, evaluando la situación actual de este proyecto, se determinó la duración del sprint en un rango de entre 15 y 20 días laborables, cada día conformado por 8 horas de trabajo, como se muestra en el gráfico anterior.

#### **4.1.10. Distribución de los User Story**

Scrum busca optimizar los tiempos de entrega evitando todo tipo de retraso y haciendo un uso efectivo de sus recursos, para ello el siguiente paso es la distribución balanceada de los User Story o requerimientos para cada Sprint o ciclo de trabajo establecido.

Quienes se encargan de realizar esta distribución son el Scrum Master y el Team Scrum, teniendo como base el balance equitativo de esfuerzo y recursos para cada Sprint. Cabe resaltar que la distribución solo se aplica para los User Story Funcionales, debido a que el desarrollo y cumplimiento de los User Story No Funcionales debe ser tomado en cuenta durante todo el desarrollo del proyecto.

▪ **Sprint Backlog 1:**

- Abstracción y recopilación de datos del lenguaje de señas.
- Definición de las letras del abecedario.
- Toma de fotos de las distintas posiciones de la mano.

En el inicio del proyecto, se evita recargar el Sprint con mucho trabajo, por ello se designa los requerimientos más livianos, es decir los User Story que tienen menos Story Point, ya que se requiere de mayor consumo de tiempo y recursos para la obtención de las imágenes que se requieren procesar en la investigación.

▪ **Sprint Backlog 2:**

- Elección de los algoritmos.
- Elección del lenguaje de programación

Para este grupo se asigna User Story con puntuaciones moderadas, de esta forma se logra equilibrar el trabajo con respecto al primer sprint, ya que en este ciclo no se va a destinar ni tiempo ni recursos para los procesos iniciales.

- **Sprint Backlog 3:**

- Procesamiento de las imágenes.

En este grupo, si bien se asigna solo un User Story, este tiene el puntaje más alto en Story Points, lo que permite una distribución equitativa de los Sprints.

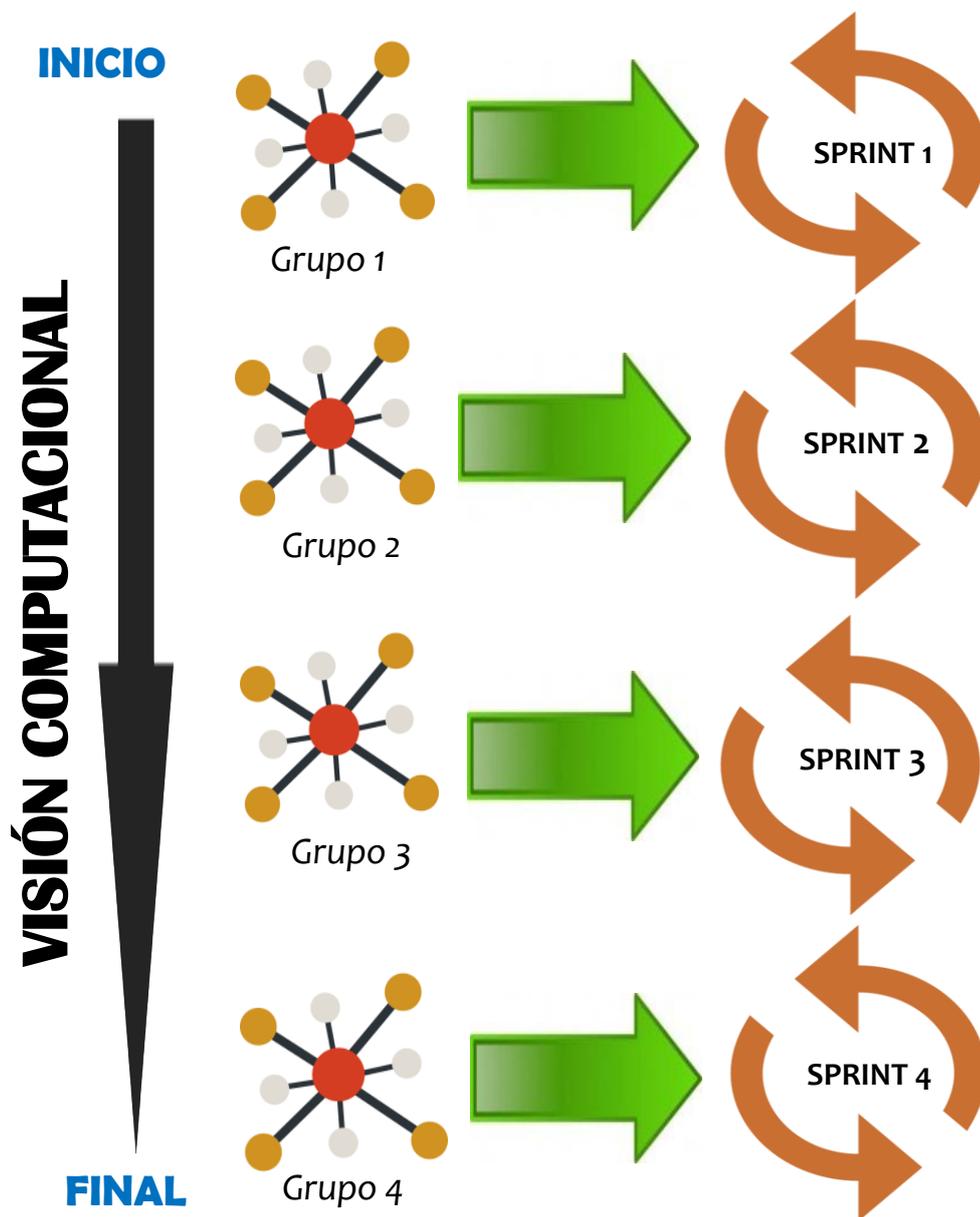
- **Sprint Backlog 4:**

- Entrenamiento del modelo de traducción.
- Desarrollo de la interfaz del modelo de traducción.
- Desarrollo del Seleccionador del lenguaje de señas.

En este último grupo, se asignan los User Story restantes, los cuales tienen Story Points moderados, debido a que la investigación se acerca a su etapa final.

#### **4.1.11. Definición de Sprints**

Realizada la distribución de los grupos de forma equitativa, donde se balanceo el consumo de tiempo y recursos, se obtiene un total de tres Sprints que se van a desarrollar a lo largo del proyecto, los cuales deben cumplir con cada uno de los User Story asignados.



*Ilustración N° 22: Total de Sprints obtenidos.*

#### **4.1.12. Duración de la Fase de Desarrollo**

En base a los Sprints obtenidos y teniendo en cuenta la duración de cada Sprint, es posible determinar la duración de la fase de desarrollo del Proyecto de Investigación, tiempo en el cual el equipo debe cumplir con los entregables periódicos.

- **Total de Sprints** : 4
- **Duración de cada Sprint** : 20 días laborables

***4 Sprints x 20 Días = 80 días de duración***

Teniendo en cuenta que se toman los 20 días laborables por mes, cada Sprint sería equivalente a 1 mes, lo cual quiere decir que la duración de la fase de desarrollo del proyecto sería de 4 meses, equivalente a **80 días laborables**.

#### 4.1.13. Cronograma

Se establecen los tiempos para las etapas del desarrollo del proyecto.

*Tabla N° 7: Cronograma de actividades.*

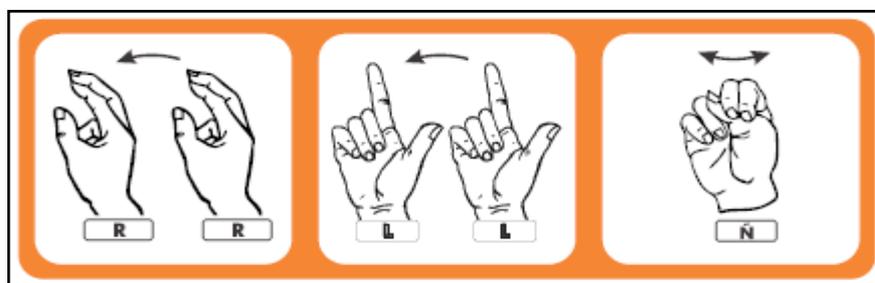
N°	ACTIVIDAD	FECHA DE INICIO	FECHA DE TERMINO
1	Recolección de requerimientos	16/10/2017	20/10/2017
2	Análisis de requerimientos	23/10/2017	03/11/2017
3	Diseño del proyecto	06/11/2017	17/11/2017
4	Desarrollo del Sprint 1	20/11/2017	15/12/2017
5	Desarrollo del Sprint 2	18/12/2017	12/01/2018
6	Desarrollo del Sprint 3	15/01/2018	09/02/2018
7	Desarrollo del Sprint 4	12/02/2017	09/03/2018
8	Pruebas y entrega final	12/03/2018	06/04/2018

## 4.2. Definición de Parámetros

### 4.2.1. Alfabeto Dactilológico Peruano

El alfabeto dactilológico peruano consta de 26 letras individuales que son representadas con una forma específica, y en algunos casos con un movimiento específico, y 3 letras dobles que requieren de un movimiento adicional a la forma que ya se muestra con la letra individual.

Las letras dobles se ejecutan haciendo un movimiento corto en línea recta hacia la derecha, este es el caso de las letras “RR” y “LL”, en el caso de la “Ñ”, se ejecuta con un doble movimiento hacia ambos lados.



*Ilustración N° 23: Letras dobles del alfabeto dactilológico peruano*

**Fuente:** (Ministerio de Educación, Peruano, 2010)

### 4.2.2. Definición de letras de abecedario

Como se detalla en el alcance de esta investigación, se va a realizar la traducción de las letras del alfabeto del lenguaje de señas que utilizan

solo una mano y no requieren de un movimiento adicional para la formación de la letra.

Por lo tanto, de las 29 letras del alfabeto dactilológico peruano, 24 letras son seleccionadas para la traducción del Lenguaje de Señas, estas letras son las siguientes:



*Ilustración N° 24: Alfabeto seleccionado del Lenguaje de Señas Peruano.*

#### **4.2.3. Toma de fotos de las posiciones de la mano**

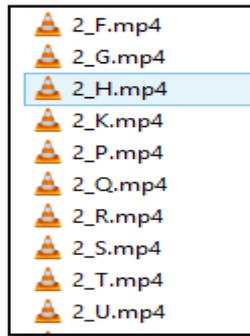
Para el desarrollo de esta investigación, se grabó de forma individual a 4 personas mientras representaban con la mano las letras del alfabeto seleccionado; posteriormente, de estos videos se extrajeron 100 imágenes con la posición exacta de la mano por cada letra representada.

$$4 \text{ personas} \times 24 \text{ letras} \times 100 \text{ imágenes} = 9600 \text{ imágenes}$$

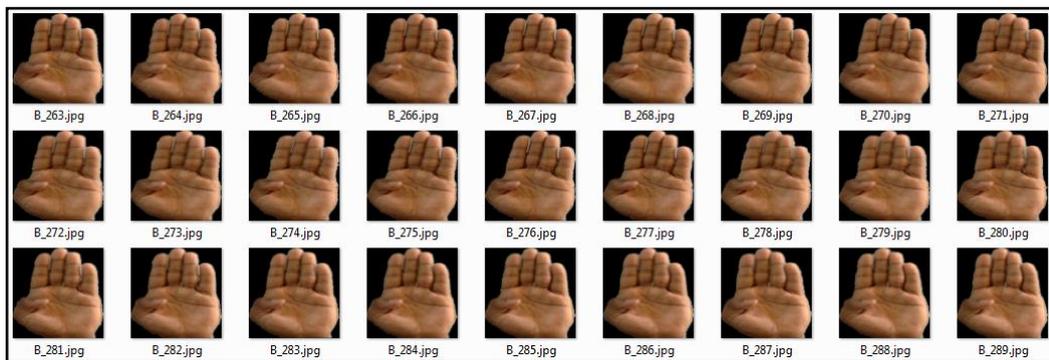
Por lo tanto, se va a procesar un total de 9600 imágenes, con un tamaño de 200\*200 pixeles y fondo de color negro, esto con la finalidad de que las imágenes para el entrenamiento sean de peso liviano, permitiendo que el modelo de traducción del lenguaje de señas pueda procesar una gran cantidad de imágenes.



*Ilustración N° 25: Número de personas para la obtención de imágenes*



*Ilustración N° 26: Videos de la representación del alfabeto seleccionado*



*Ilustración N° 27: Imágenes por cada letra.*

#### **4.2.4. Elección de algoritmos**

Dado que esta investigación busca lograr la traducción del lenguaje de señas a través de un modelo que haga uso de la visión computacional, se determina que el proceso de desarrollo cumple con las características propias de un aprendizaje supervisado, donde el objetivo son las predicciones basadas en características o comportamientos obtenidos de datos almacenados, en este caso las 9600 imágenes, buscando patrones y relacionando todos los campos con el campo objetivo, es decir la letra que representa la mano en cada imagen (González, 2014).

## **Para el procesamiento de imágenes**

Se ha seleccionado los siguientes algoritmos:

- a. **Filtro Gaussiano**, utilizado en la eliminación del ruido de las imágenes.
- b. **Operador laplaciana**, necesario para el realce de bordes de las imágenes.
- c. **Canny**, empleado para la detección de los bordes de la mano.

El desarrollo del procesamiento de las imágenes se detalla más adelante, además se muestra los resultados obtenidos con la aplicación de otros algoritmos y la comparación con el resultado que se obtiene con los algoritmos ya seleccionados, esto con la finalidad de demostrar mediante ejemplos el motivo de esta elección.

## **Para el entrenamiento del modelo**

Para explicar la elección del algoritmo para el entrenamiento del modelo de traducción del lenguaje de señas, es necesario explicar en qué consisten las Máquinas de Vectores de Soporte, cuyas siglas en inglés son SVM (Support Vector Machines).

- Las SVM son modelos de aprendizaje supervisados que cuentan con algoritmos de aprendizajes asociados, cuya función es analizar los datos empleados en la clasificación y el análisis de regresión. Este sistema es empleado para el entrenamiento de máquinas de aprendizaje lineal, logrando resultados eficientes tanto en la

clasificación como en la regresión (análisis de variables), entre las aplicaciones que este modelo están la clasificación de imágenes, el reconocimiento de caracteres, identificación de funciones, entre otros (Resendiz Trejo, 2006).

Por lo tanto, se elige uno de los algoritmos más representativos de las SVM para la fase de entrenamiento de esta investigación, el algoritmo seleccionado es el Histograma de Degradados Orientados (HOG).

**d. HOG**, es un algoritmo de extracción de características, también llamado descriptor de características, donde la información de cada pixel de la imagen se convierte en un vector o conjunto de características.

Este algoritmo utiliza la distribución (histogramas) de las direcciones de los gradientes (gradientes orientados) como características, donde los gradientes de una imagen son útiles porque la magnitud de los gradientes es grande alrededor de los bordes y esquinas, esto se debe a que en estas regiones se presentan los cambios abruptos de intensidad. Además, como parte del entrenamiento con este algoritmo, es necesario procesar la imagen a evaluar, a través del uso del filtro Sobel en los ejes X y Y.

#### **4.2.5. Elección del lenguaje de programación**

Para el desarrollo del modelo se ha optado por hacer uso del lenguaje de programación Java, debido a las siguientes razones:

- Es uno de los lenguajes más conocidos y robustos que existen en el mercado.
- Actualmente tiene mayor soporte para las librerías de OpenCV, contando con páginas web dedicadas a documentar la programación en Java con OpenCV.
- Cuenta con un entorno visual mucho más desarrollado que otros lenguajes de programación, lo que permite revisar de forma gráfica el avance de la programación (ventanas, botones, etc).

### 4.3. Reconocimiento de patrones

#### 4.3.1. Correlación Normalizada

Este método de reconocimiento de patrones no requiere un procesamiento previo de la imagen, salvo el recorte de un patrón específico; la correlación consiste en la localización de un patrón dentro de una imagen, para lo cual se requiere de una imagen general y el recorte de un patrón que se encuentra dentro de la imagen principal. (Escalera, 2001)

La correlación  $g(x)$  entre dos funciones  $f(x)$  y  $h(x)$  es:

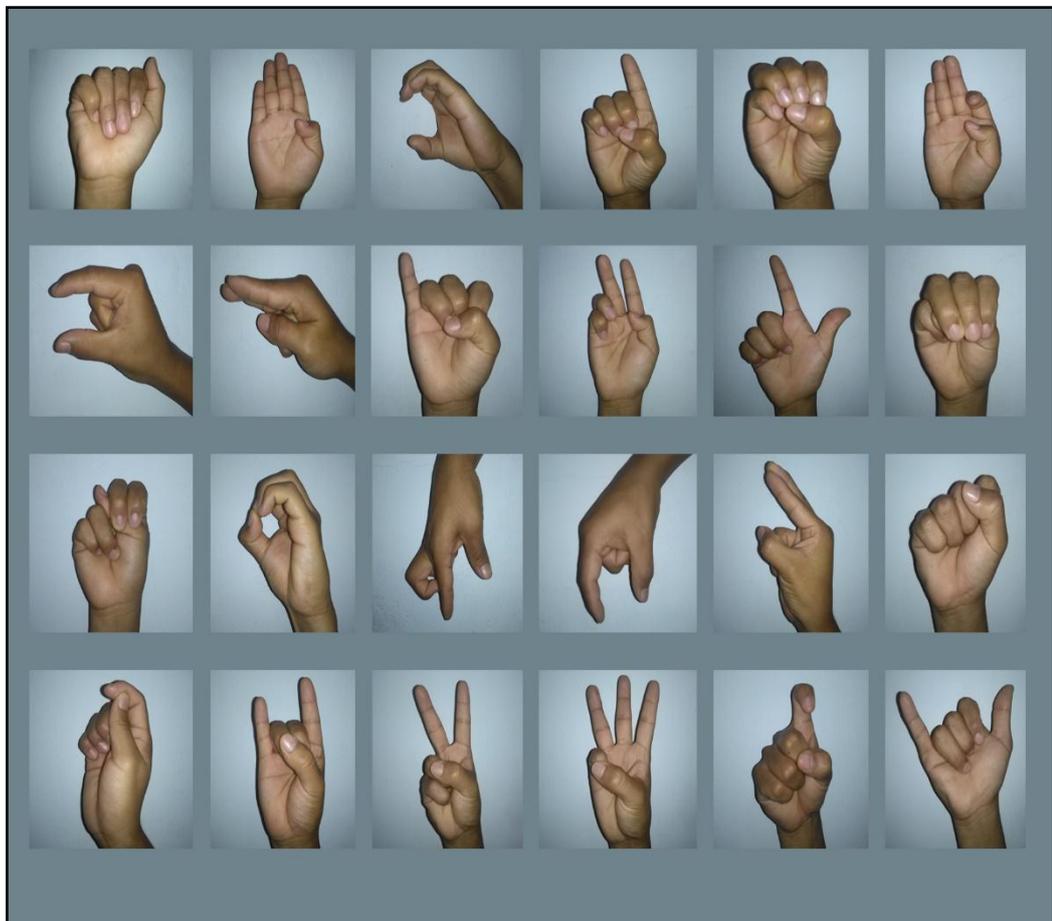
$$g(x, y) = h(x, y) \circ f(x, y) = \sum_{i=-\infty}^{i=\infty} \sum_{j=-\infty}^{j=\infty} f * (i, j) h(x + i, y + j)$$

La ventaja de este método es su rapidez debido a que no requiere procesamientos previos de la imagen, sin embargo, este proceso no es invariante a la rotación ni a la escala, es decir si la resolución del

patrón es reducida o aumentada, el reconocimiento no será efectivo; lo mismo sucede con la rotación. Esto se debe a que es necesario que cada pixel coincida entre el patrón y la imagen.



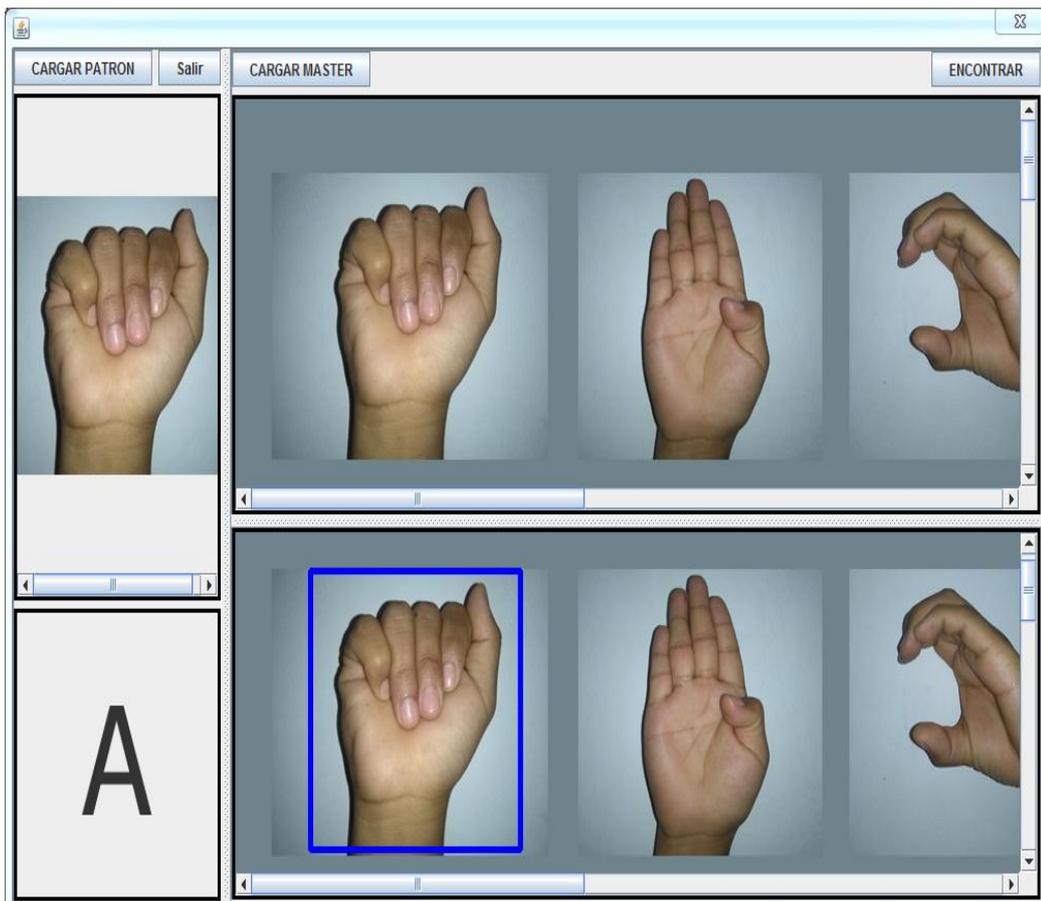
*Ilustración N° 28: Ejemplo 1- Correlación de imágenes.*



*Ilustración N° 29: Ejemplo 2- Imagen general.*



*Ilustración N° 30: Ejemplo 2- Patrón*



*Ilustración N° 31: Ejemplo 2- Patrón localizado por correlación.*

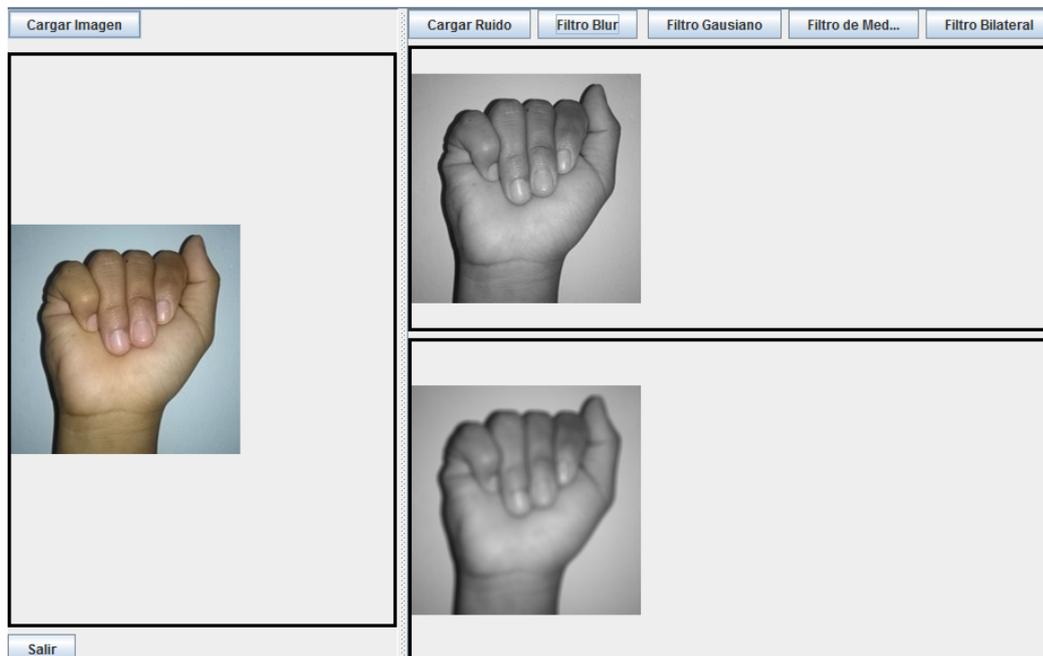
## 4.4. Procesamiento de imágenes

### 4.4.1. Eliminación de ruido

El ruido consiste en valores distorsionados en la imagen, que se manifiestan generalmente en píxeles aislados que toman un valor de gris diferente al de sus vecinos (Escalera, 2001).

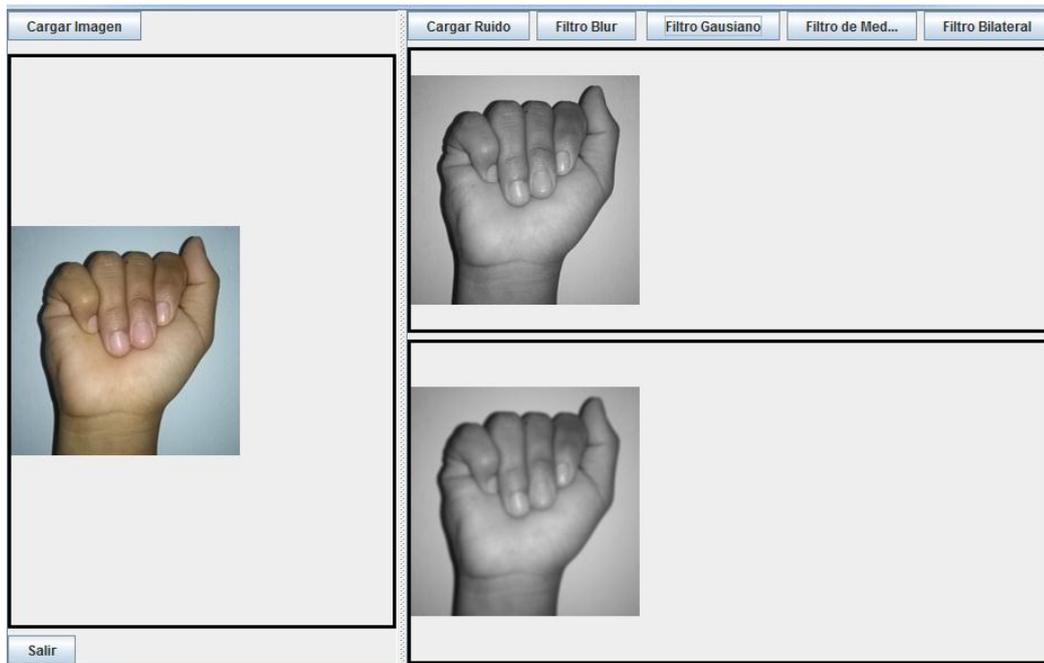
Para eliminar estas distorsiones que podrían perjudicar el realce de los bordes de la imagen procesada, se utilizaron los siguientes filtros:

- Filtro de Media (Blur)



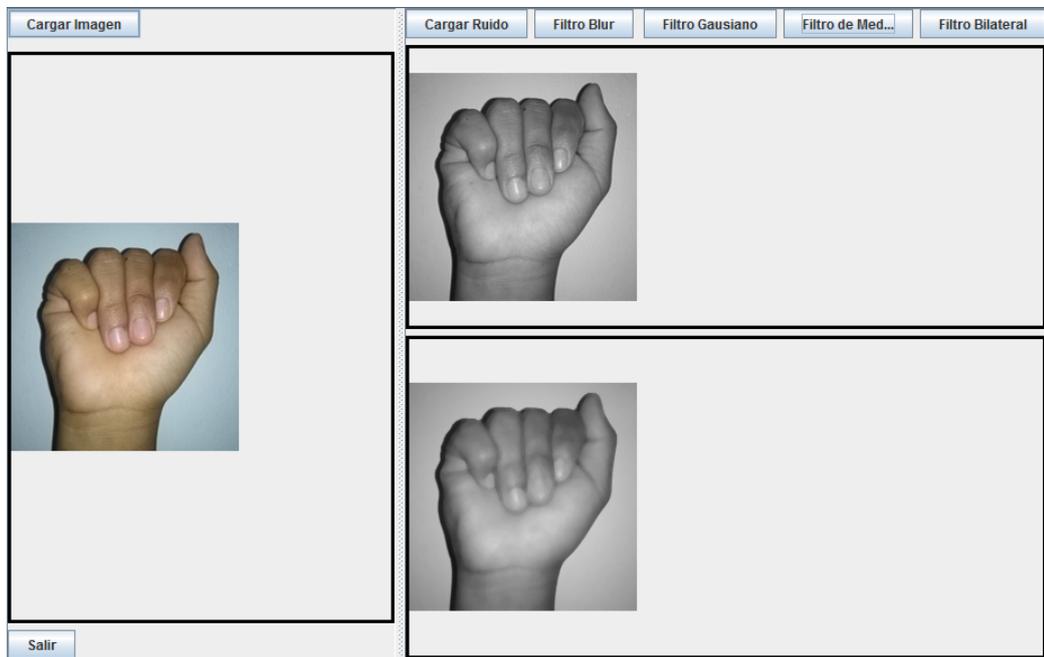
*Ilustración N° 32: Aplicación de filtro de Media.*

- Filtro Gaussiano



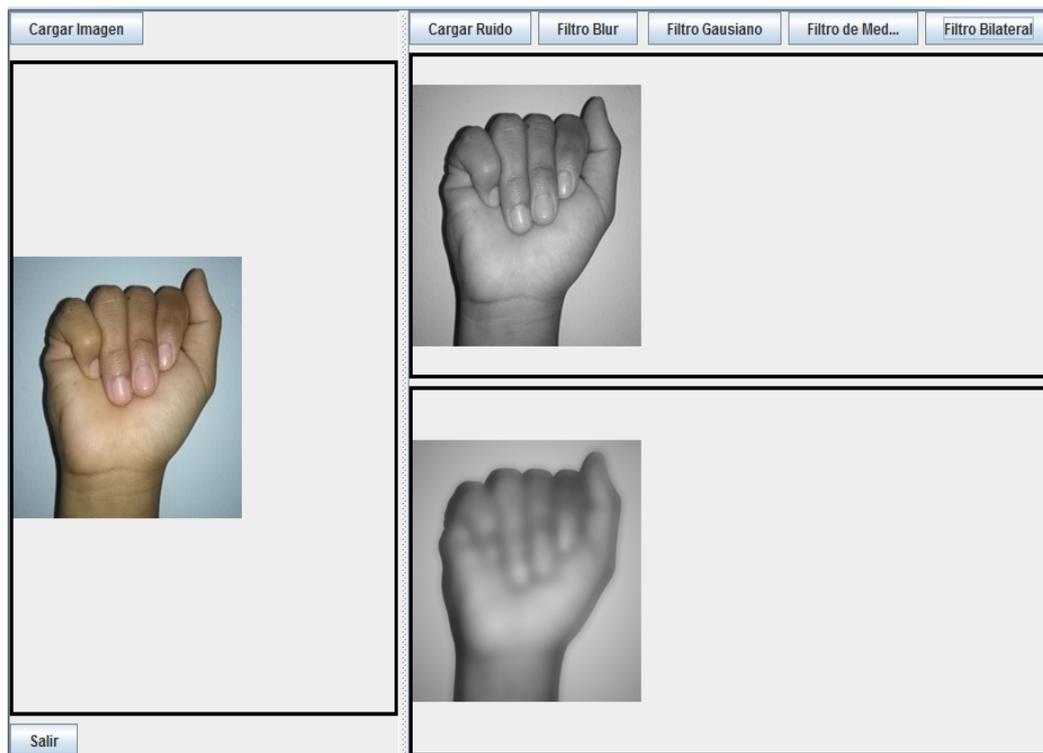
*Ilustración N° 33: Aplicación de filtro Gaussiano.*

- Filtro de Mediana



*Ilustración N° 34: Aplicación de filtro de Media.*

- Filtro Bilateral



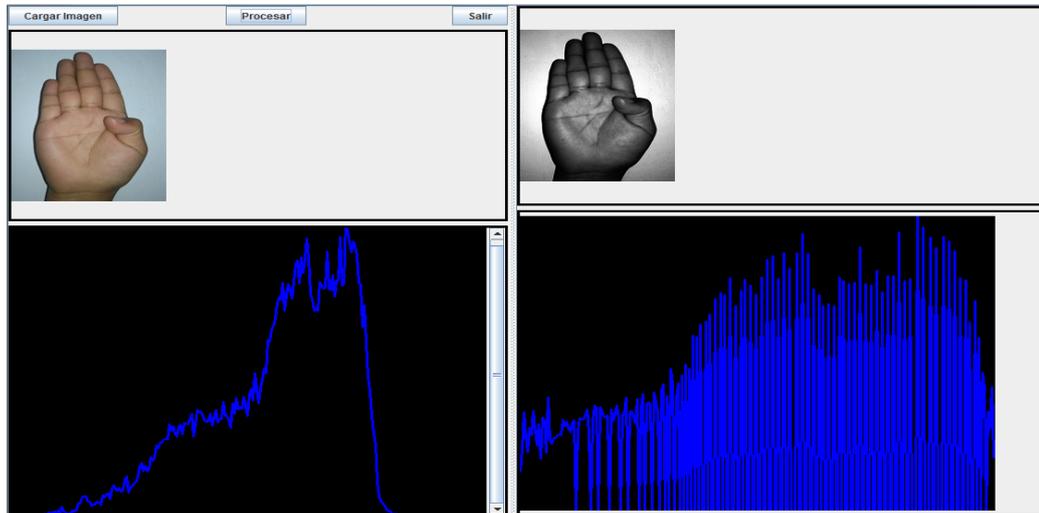
*Ilustración N° 35: Aplicación de filtro Bilateral.*

Se elige el Filtro Gaussiano debido a que tiene un mejor resultado y requiere poco procesamiento lo que implica un menor uso de los recursos computacionales. Como se puede apreciar en las imágenes, el filtro de Mediana arroja un mejor resultado en cuanto a la nitidez de la imagen, sin embargo, este filtro requiere mayor uso de recursos y mayor tiempo de procesamiento lo cual va en contra de los objetivos de la investigación, es por eso que se escoge el filtro Gaussiano, ya que va a permitir realizar un procesamiento mucho más rápido.

#### **4.4.2. Balanceo de contraste**

Para tener una información global de toda la imagen, es necesario analizar y modificar el histograma, para ello se realiza el balanceo de contraste, también llamado ecualización del histograma, con lo cual se

pretende que para todos los valores de gris se obtenga el mismo número de píxeles (Escalera, 2001).



*Ilustración N° 36: Aplicación de balanceo de contraste.*

#### **4.4.3. Realce de bordes**

El realce de bordes logra un resultado opuesto a la eliminación de ruido, ya que en este punto se busca resaltar los píxeles que presentan un valor de gris distinto al de sus vecinos. Si la imagen es ruidosa, el efecto del ruido se multiplicará con este proceso; por lo que es necesario eliminar el ruido antes de resaltar los bordes (Escalera, 2001).

Realce de bordes con un umbral

- Filtro 1: Umbral de operador de media

Este filtro se basa en la eliminación o disminución de las bajas frecuencias.

1	1	1
---	---	---

1	1	1
1	1	1

*Ilustración N° 37: Matriz para operador de media.*

Para este filtro, la variación del número central de la matriz permite dar mayor realce a los contornos de la imagen.

- Filtro 2: Umbral de operador laplaciana

Este filtro busca dar más valor a las altas frecuencias, es decir realzar los cambios bruscos del nivel de gris.

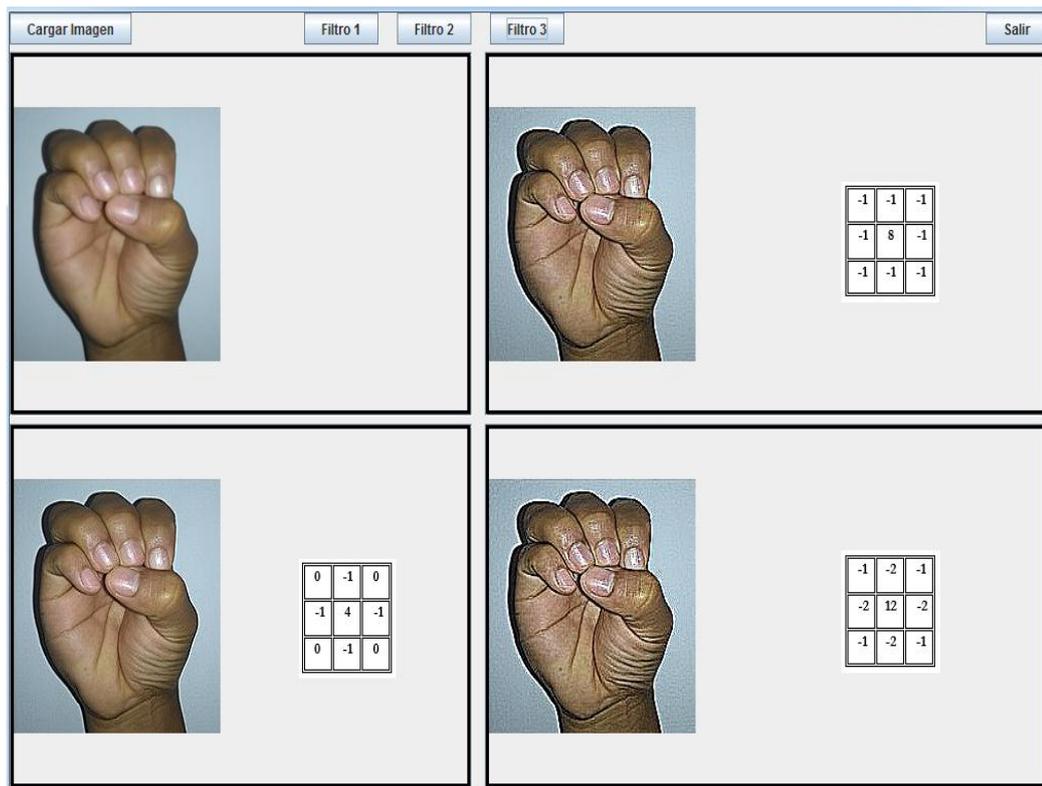
Como se muestra en las siguientes imágenes, se comprueba que las matrices o convoluciones se aproximan a tener el mismo valor para todas las direcciones.

-1	-1	-1
-1	8	-1
-1	-1	-1

0	-1	0
-1	4	-1
0	-1	0

-1	-2	-1
-2	12	-2
-1	-2	-1

*Ilustración N° 38: Matrices de operador laplaciana.*



*Ilustración N° 39: Aplicación del realce de bordes con umbral de operador laplaciana.*

El filtro que se utiliza para este punto, es el umbral de operador laplaciana, ya que este expresa la derivada en todas las direcciones.

#### **4.4.4. Detección de bordes**

Las técnicas utilizadas en la detección de bordes tienen como objetivo la localización de los puntos donde se produce una variación de intensidad, es decir detecta los cambios bruscos de intensidad de gris dentro de una imagen, empleando para ello métodos basados en los operadores derivada (Escalera, 2001). A partir de este punto, es necesario que las imágenes sean convertidas a escala de grises.

Los algoritmos empleados en el desarrollo de esta investigación, fueron el Algoritmo Sobel y el Algoritmo Canny.

#### 4.4.4.1. Algoritmo Canny

Para la detección de bordes con este algoritmo es necesario la optimización de las siguientes condiciones:

- ✓ Solo deben ser detectados los bordes.

$$Error = SNR = \frac{A \left| \int_{-W}^0 f(x) dx \right|}{\sqrt[n_0]{\int_{-W}^W f^2(x) dx}}$$

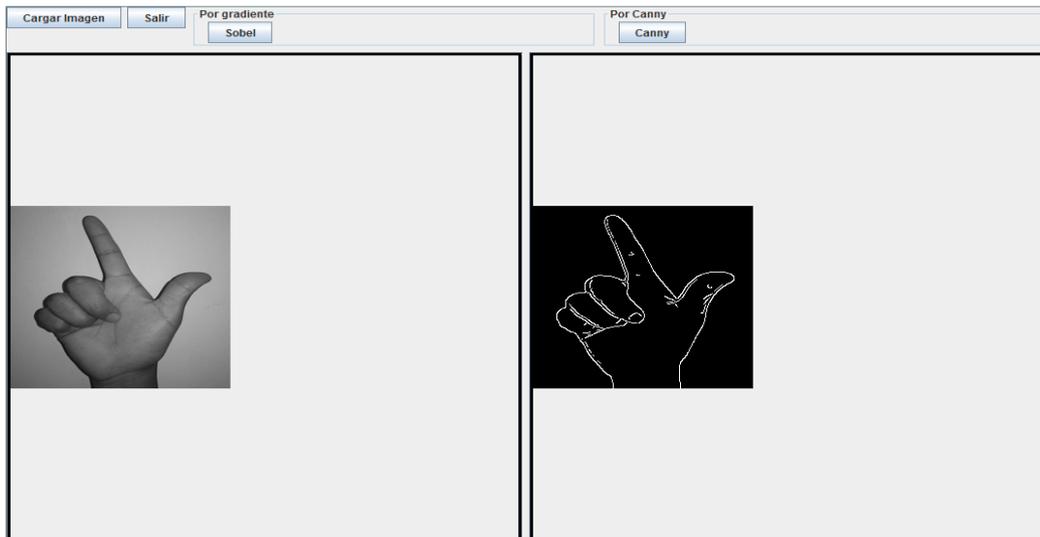
- ✓ Mínima distancia entre el pixel señalado como borde y el borde real.

$$Localización = \frac{A|f(0)|}{\sqrt[n_0]{\int_{-W}^W f^2(x) dx}}$$

- ✓ Cuando solo exista un pixel como borde, no deben identificarse más pixeles.

$$Distancia = \pi \left( \frac{\int_{-\infty}^{\infty} f^2(x) dx}{\int_{-\infty}^{\infty} f'^2(x) dx} \right)^{\frac{1}{2}}$$

Canny busca optimizar el producto que se obtiene de la relación señal ruido por la localización y una condición, llegando a la conclusión que el operador óptimo es la derivada de una gaussiana. (Escalera, 2001).



*Ilustración N° 40: Aplicación del Algoritmo Canny*

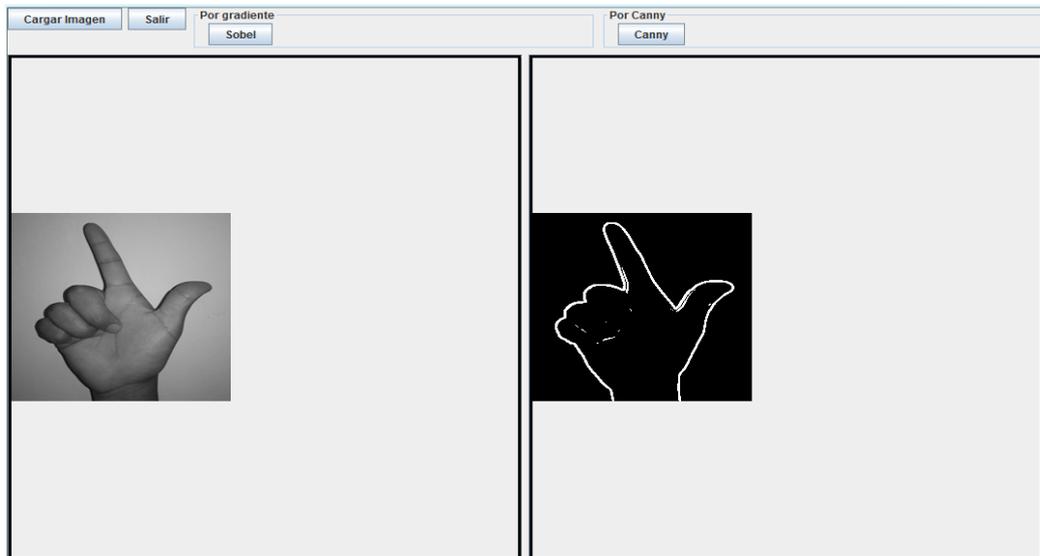
#### 4.4.4.2. Algoritmo Sobel

Está basado en el operador Prewitt, el cual expande la definición de gradiente a un entorno de 3x3 para ser más inmune al ruido, pero el algoritmo sobel a diferencia del prewitt, da más importancia a los pixeles centrales (Escalera, 2001).

-1	0	1
-2	0	2
-1	0	1

-1	-2	-1
0	0	0
1	2	1

*Ilustración N° 41: Matrices de operador sobel.*



*Ilustración N° 42: Aplicación de Algoritmo Sobel.*

El algoritmo elegido para esta investigación es el algoritmo Canny, ya que como se puede apreciar en las Ilustraciones N° 36 y 37, se obtiene un mejor resultado en la detección de los bordes con este algoritmo.

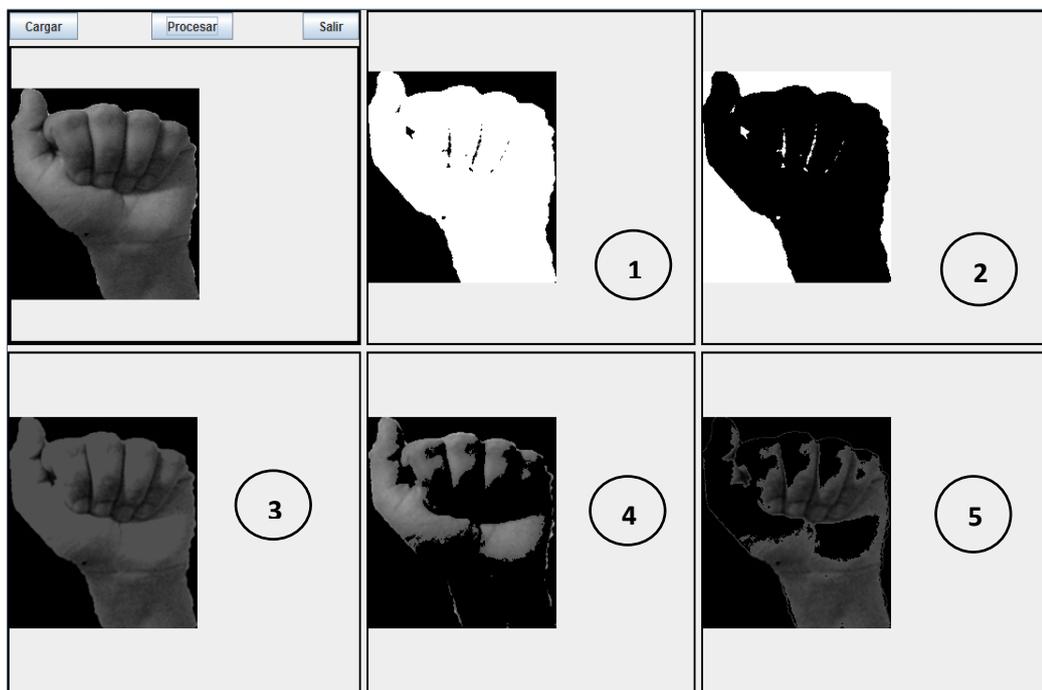
#### **4.4.5. Segmentación**

El objetivo es separar los objetos del medio en el que se encuentran y distinguirlos entre sí, para lo cual hace uso de algunas propiedades como los niveles de gris, color, textura, bordes o movimiento.

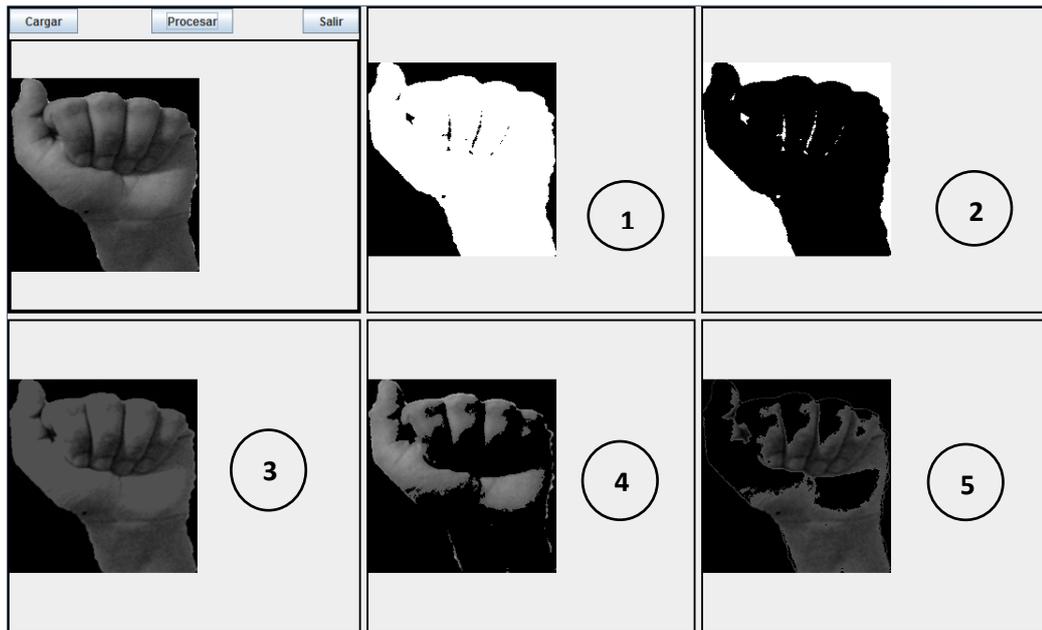
La segmentación se basa en tres propiedades: similitud, donde cada pixel de un elemento tiene valores parecidos para alguna propiedad; discontinuidad, es decir cuando los objetos tienen bordes definidos; y conectividad, donde los pixeles pertenecientes al mismo objeto deben estar agrupados (Escalera, 2001). Además, para segmentar una imagen se requiere del proceso de umbralización.

Los umbrales utilizados en los ejemplos son los siguientes:

1. THRESH\_BINARY
2. THRESH\_BINARY\_INV
3. THRESH\_TRUNC
4. THRESH\_TOZERO
5. THRESH\_TOZERO\_INV



*Ilustración N° 43: Ejemplo 1- Segmentación con umbrales*



*Ilustración N° 44: Ejemplo 2 - Segmentación con umbrales*

#### **4.4.5.1. Binarización**

Consiste en una reducción de información donde los únicos valores posibles son verdadero y falso (1 y 0), en el caso de una imagen digital, estos valores corresponden a dos colores: blanco y negro (Escalera, 2001).

Para realizar este proceso con las librerías opencv es necesario un umbral, para que los valores de los pixeles que sean menores al umbral sean iguales a 0 y los valores mayores al umbral sean iguales a 1.

En el desarrollo de esta investigación, se puede apreciar que en las Ilustraciones N° 39 y N° 40, se obtiene mejores resultados a través del uso de la binarización con los

umbrales THRESH\_BINARY y THRESH\_BINARY\_INV para el proceso de segmentación, ya que define mejor el área de la mano, logrando detectar su posición.

#### **4.4.6. Transformaciones morfológicas**

Terminado el proceso de segmentación, es necesario evaluar las imágenes resultantes, donde en muchos casos se presentan problemas debido a una mala clasificación de los píxeles; por ejemplo, cuando un píxel es tomado como objeto de estudio y no como parte del fondo, lo cual distorsiona el resultado del procesamiento de la imagen.

Para resolver este problema, las transformaciones morfológicas modifican la estructura de los objetos presentes en la imagen, además de permitir la eliminación de ruido que se produce en el proceso de segmentación (Escalera, 2001).

Las transformaciones morfológicas que se aplican para esta investigación, son las siguientes:

##### **4.4.6.1. Erosión**

Significa la degradación progresiva de uno de los campos de la imagen, si un elemento del campo está rodeado de elementos iguales a él, no se modificará, caso contrario pasará a formar parte del otro campo (Escalera, 2001).

En este proceso la imagen ya ha sido binarizada, por lo tanto, los campos 0 ó 1 van a ser modificados si se encuentra rodeado de elementos de un campo distinto al que pertenecen. Este paso se realiza con la finalidad de eliminar objetos aislados e independientes que se encuentran en la imagen y que no forman parte en la identificación de la forma de la mano.

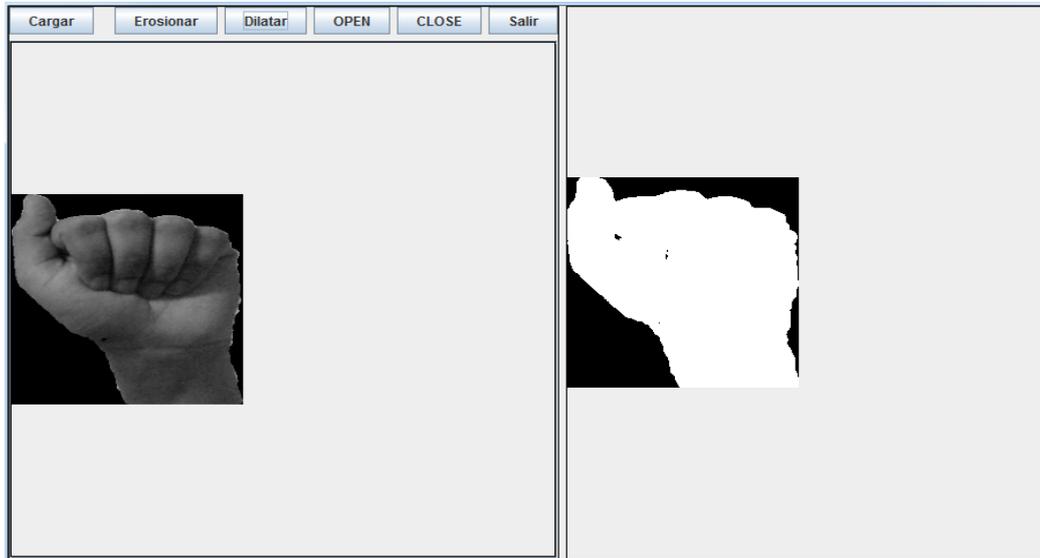


*Ilustración N° 45: Ejemplo de erosión.*

#### **4.4.6.2. Dilatación**

Es el crecimiento progresivo de uno de los campos (0 ó 1), esto es posible si alguno de los elementos vecinos pertenece al mismo campo del elemento que se desea expandir (Escalera, 2001).

Esto también va a permitir eliminar elementos aislados que se originan debido al ruido y que podría distorsionar el resultado en la traducción de imagen.

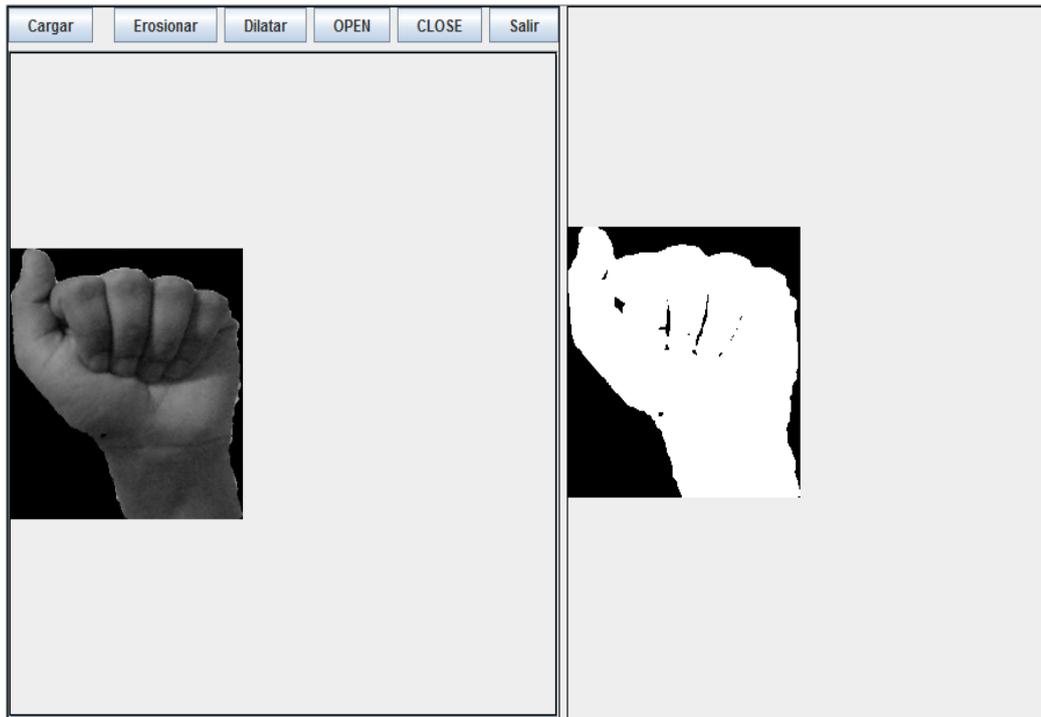


*Ilustración N° 46: Ejemplo de dilatación.*

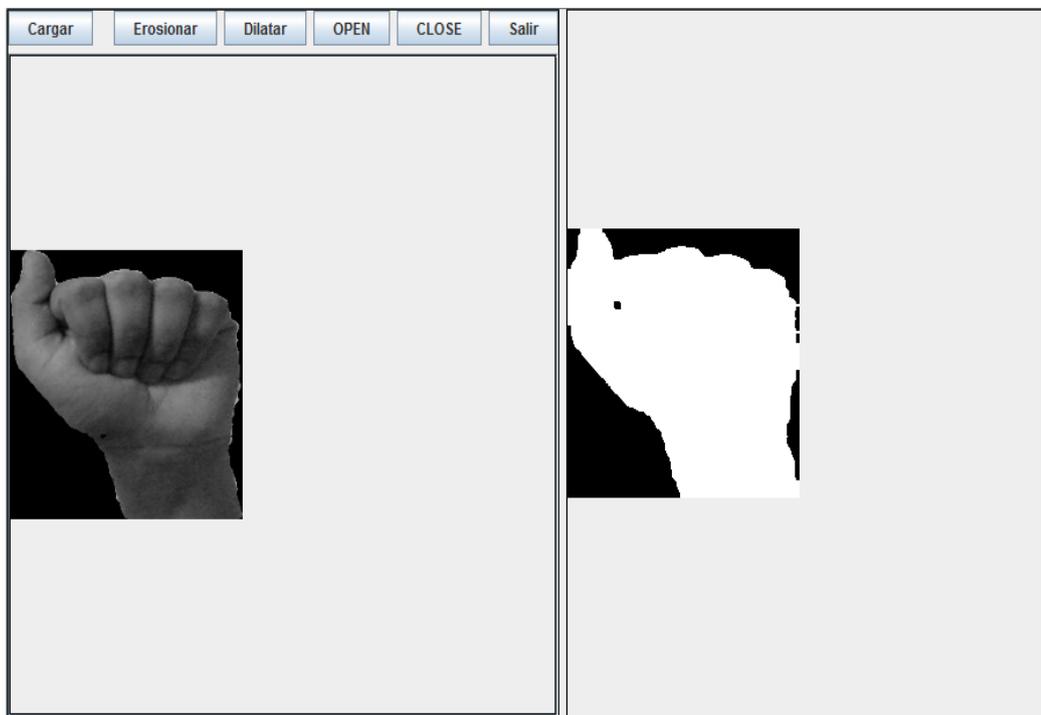
#### **4.4.6.3. Apertura y Cierre**

Si bien es cierto, las dos transformaciones antes aplicadas, logran eliminar pequeños elementos originados por el ruido, luego de una erosión no es posible recuperar la imagen original, lo mismo sucede en el caso de una dilatación.

Es por eso que se aplica la apertura y cierre, que viene a ser la combinación de las erosiones y dilataciones, donde la apertura suaviza los contornos del objeto y rompe enlaces delgados, y el cierre, suaviza contornos y rellena pequeños agujeros que se producen en la apertura.



*Ilustración N° 47: Ejemplo de apertura.*



*Ilustración N° 48: Ejemplo de cierre.*

#### **4.5. Actualización del Sprint Backlog**

Una vez terminado el desarrollo del Sprint 3, los investigadores se toparon con una dificultad que limitaba el avance al siguiente Sprint, para solucionar este problema fue necesario cambiar los parámetros ya establecido en el Sprint 2.

Si bien es cierto, se lograron resultados óptimos en la fase de procesamiento de imágenes por medio de la programación en el entorno de java, esto no fue así en la fase de entrenamiento del modelo debido a las siguientes razones:

- Poca o nula información sobre la integración o el uso de SVM en el entorno de java con las librerías opencv,
- Mayor búsqueda de información sobre documentación necesaria para el entrenamiento con lenguaje java, sin obtener resultados exitosos.
- Necesidad del uso de SVM para el desarrollo del algoritmo HOG y el posterior entrenamiento del modelo.
- Limitación para futuras mejoras de esta investigación, debido a la falta de documentación que explique cómo aplicar el aprendizaje automático en un entorno java.

Con la finalidad de dar solución a este problema y continuar con el desarrollo del entrenamiento del modelo, los investigadores optaron por cambiar el lenguaje de programación java por otro lenguaje que cuente con documentación necesaria para la integración de SVM, para ello fue necesario actualizar el Sprint Backlog 4.

En este último Sprint, se considera la elección del lenguaje de programación y el procesamiento de imágenes, este último se desarrolla nuevamente ya que todas las funciones empleadas con OpenCV deben ser implementadas con el nuevo lenguaje de programación.

#### **4.5.1. Sprint Backlog 4**

- Nueva elección del lenguaje de programación.
- Procesamiento de imágenes.
- Entrenamiento del modelo de traducción.
- Desarrollo de la interfaz del modelo de traducción.
- Desarrollo del Seleccionador del lenguaje de señas.

Con esta actualización, el desarrollo del procesamiento de imágenes cambió totalmente debido a la elección de otro lenguaje de programación; estos cambios también permitieron que los investigadores hallaran nueva información que les brindó una perspectiva diferente sobre los métodos empleados en el procesamiento de imágenes.

Por último, los investigadores consideraron acertado plasmar en el informe final cada paso del procesamiento de imágenes con el lenguaje java, debido a que los resultados obtenidos en ese proceso fueron óptimos y esta documentación puede servir como base para futuras investigaciones donde ya se cuente con información para aplicar SVM en un entorno java.

#### **4.5.2. Nueva elección de lenguaje**

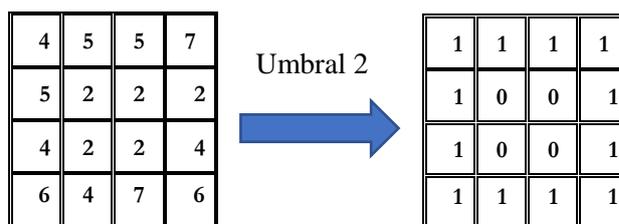
El nuevo lenguaje de programación seleccionado para esta investigación es el lenguaje Python, debido a que cuenta con amplia información para realizar la integración de las SVM con librerías Opencv.

Cabe mencionar que el cambio de lenguaje de programación no significó un retraso importante para la investigación, debido a que las funciones de Opencv no varían de un lenguaje a otro, lo que disminuye el tiempo de aprendizaje para el desarrollo con el lenguaje Python, por lo que fue suficiente el conocimiento esencial de los investigadores sobre el comportamiento de las funciones y el resultado que se muestra con las librerías de Opencv para continuar con el desarrollo de esta investigación.

#### **4.5.3. Nuevo procesamiento de imágenes**

Si bien es cierto, en la programación con el lenguaje java se procesaron imágenes en escala de grises y el método de binarización con el uso de umbrales dio un buen resultado en el proceso de segmentación, los investigadores no tomaron en cuenta las diferentes tonalidades de piel y la variación en la luminosidad del entorno donde se encuentra la mano.

Considerando el tono de piel y la luminosidad del entorno, se establece un nuevo proceso donde la segmentación es aplicada en imágenes a color y no en escala de grises. Para demostrar esto se plantea el siguiente ejemplo:



*Ilustración N° 49: Aplicación de umbral en matriz de imagen.*

Como se puede apreciar en la Ilustración N° 50, para la elección del valor de un umbral, es necesario conocer el valor de la intensidad de color de la imagen que se quiere segmentar, además la iluminación debe tener un valor constante para cada ambiente, caso contrario se alteraría los valores en la binarización.

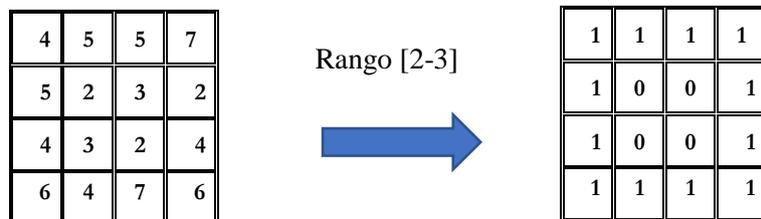
A través de este ejemplo se demuestra que no es viable el procesamiento de imágenes en escala de grises ni la aplicación de umbrales para la segmentación; por lo tanto, se opta por emplear otro espacio de color alternativo al BGR, con la finalidad de que la investigación se adapte al entorno y a los diferentes tonos de piel de las manos a procesar para el reconocimiento del lenguaje de señas.

#### **4.5.3.1. Espacio de color alternativo**

Uno de los espacios de color más conocido es el HSI, el cual se basa en la simulación de la visión humana para la percepción de los colores, para ello caracteriza el color en términos de tono (Hue), saturación (Saturation) y brillo (Intensity), permitiendo realizar la segmentación de la imagen en base al tono o tinte del color (Escalera, 2001).

Una de las variantes del HSI, es el YCbCr, este espacio de color se basa en la obtención de la luminancia (luminosidad) y la crominancia (color). En las librerías de OPENCV, la imagen se convierte de valores BGR a YCbCr.

Para esta investigación se opta por escoger el espacio de color YCbCr, ya que permite la utilización de un rango de valores en reemplazo del uso de un umbral para la binarización de la imagen, en este rango se establece los valores de las variaciones del color de piel que existen, logrando obtener una segmentación más exacta de la mano, donde la luminosidad del entorno no afecte en gran medida los valores resultantes. Para representar este proceso, se plantea el siguiente ejemplo:

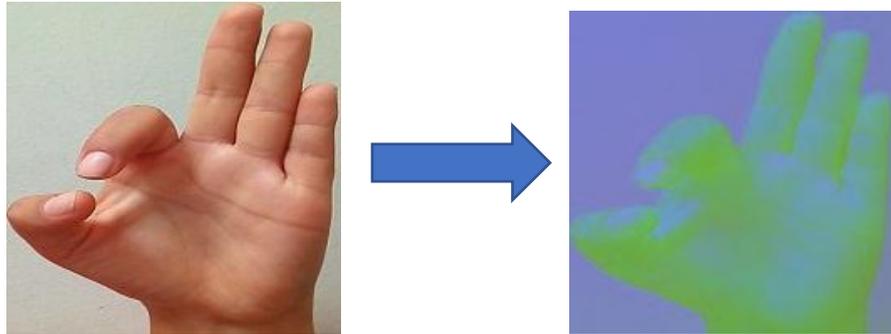


*Ilustración N° 50: Aplicación de rango de valores.*

En base a lo mencionado en el punto anterior, el nuevo procesamiento de imágenes se aplica de la siguiente manera:

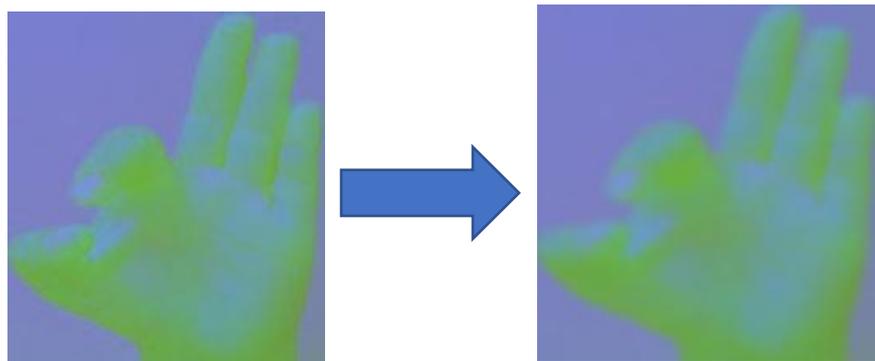
- **Conversión al espacio de color YCR\_CB:** En las imágenes a procesar, la mano representa la zona de interés para la investigación y a través de la aplicación del espacio de color

YCr\_CB, esta zona adquiere un tono verdusco que lo diferencia del tono oscuro del fondo de la imagen. Este cambio de espacio de color se muestra en las siguientes imágenes:



*Ilustración N° 51: Aplicación del espacio de color YCr-CB.*

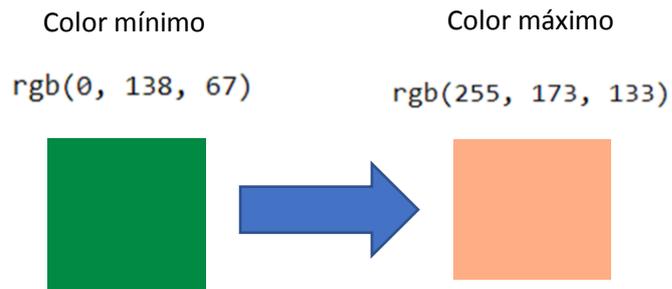
- **Eliminación de ruido (Filtro Gaussiano):** Con la imagen convertida al nuevo espacio de color se procede con la eliminación del ruido a través del Filtro Gaussiano.



*Ilustración N° 52: Eliminación de ruido.*

- **Segmentación:** Antes de continuar con el proceso de segmentación, es necesario establecer el rango que se va a evaluar para encontrar el objeto de interés, la mano.

El rango de color seleccionado está entre el color rgb (255, 173, 133), como valor máximo y el color rgb (0, 138, 67), como valor mínimo. Estos colores están basados en el tono de piel de la mano antes de ser procesada y en el tono que adquiere con la aplicación del espacio de color YCR\_CB.



*Ilustración N° 53: Rango de color.*

- **Binarización:** Por último, se aplica el proceso de binarización a la imagen segmentada, como se muestra en la siguiente imagen:



*Ilustración N° 54: Imagen binarizada.*

De esta forma, la imagen esta lista para ser comparada con las diversas imágenes empleadas en el proceso de entrenamiento del modelo.

#### **4.5.4. Estado del cronograma de desarrollo**

Con la actualización del sprint backlog 4, es de suponerse que el cronograma de la investigación se vería afectado por el retraso que origina agregar dos user story en el sprint final, sin embargo los tiempo para cada fase de la investigación no sufrieron mucha alteración, debido a que el procesamiento de imágenes que se aplica con el lenguaje Python requirió de un menor tiempo para su desarrollo, lo mismo sucedió con el desarrollo de la interfaz del modelo, este último punto es detallado más adelante. Por consiguiente, la culminación del Sprint Backlog 4 se dió dentro de la fecha establecida en el cronograma de desarrollo.

#### **4.6. Traducción del Lenguaje de Señas**

La traducción del Lenguaje de Señas consta de dos componentes esenciales: los descriptores y los clasificadores, en esta investigación se utiliza el descriptor HOG (Histogram of Oriented Gradients) y el clasificador SVM (Support Vector Machines), a continuación se detalla cada uno de estos componentes:

##### **4.6.1. Entrenamiento**

En esta etapa se hace uso de las 9600 imágenes recolectadas en la toma de fotos de las diferentes posiciones de la mano, como base para el proceso de entrenamiento.

#### 4.6.2. Descriptores

Para extraer las características de las imágenes en el proceso de entrenamiento, se va utilizar los descriptores basados en el Histograma de Degradados Orientados (HOG).

- **Pasos del descriptor HOG:**

- **Detección del Gradiente:**

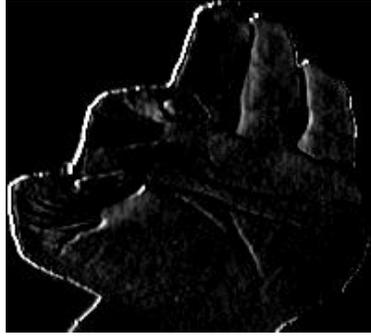
El término Gradiente representa el cambio direccional en la intensidad de la imagen, es decir, el cambio de intensidad en los píxeles de la imagen.

Para lograr la detección del gradiente, es necesario convertir la imagen a escala de grises, como se muestra a continuación:



*Ilustración N° 55: Conversión a Escala de Grises*

Luego se utiliza el filtro sobel para hallar los cambios de intensidad en los píxeles, tanto en el eje x como en el eje y de la imagen.



*Ilustración N° 56: Gradiente en dirección Horizontal (eje x)*



*Ilustración N° 57: Gradiente en dirección Vertical (eje y)*

Por último, las imágenes resultantes con la aplicación del filtro, deben ser fusionadas para obtener una sola imagen con los gradientes en los ejes horizontal y vertical. Esta fusión se realiza con la función `cv2.cartToPolar(gx, gy)`, la cual logra interpolar los valores de las dos imágenes halladas.

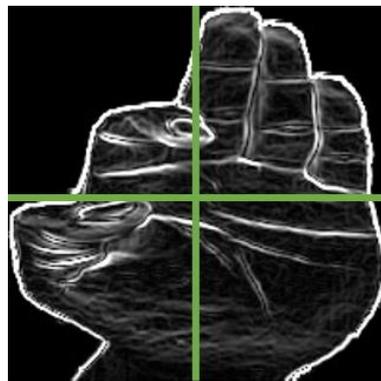


*Ilustración N° 58: Fusión de gradientes.*

- **Cálculo de los histogramas:**

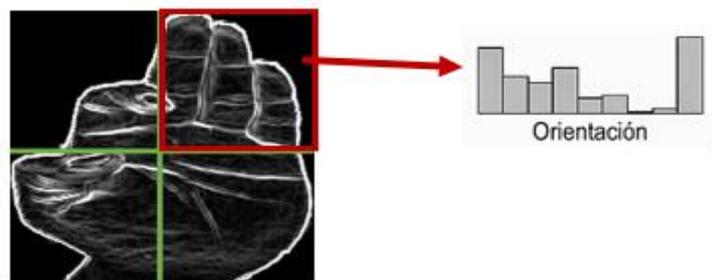
Detectado el gradiente, los siguientes pasos consisten en la división de la imagen en celdas de tamaño fijo y el cálculo de histograma de las orientaciones por cada celda.

El tamaño de las celdas es de 100x100 píxeles, teniendo en cuenta que las imágenes tienen un tamaño de 200x200 píxeles, cabe mencionar que mientras más divisiones existan en la imagen, mayor será el tiempo de procesamiento.



*Ilustración N° 59: División en celdas*

Realizada la división de la imagen en celdas, se debe calcular el histograma de orientación por cada celda resultante.



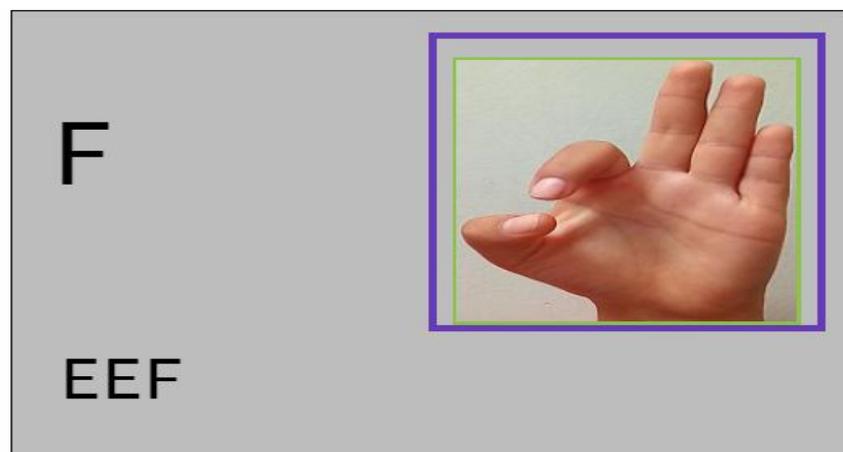
*Ilustración N° 60: Cálculo del histograma de orientación*

### 4.6.3. Interfaz del Modelo

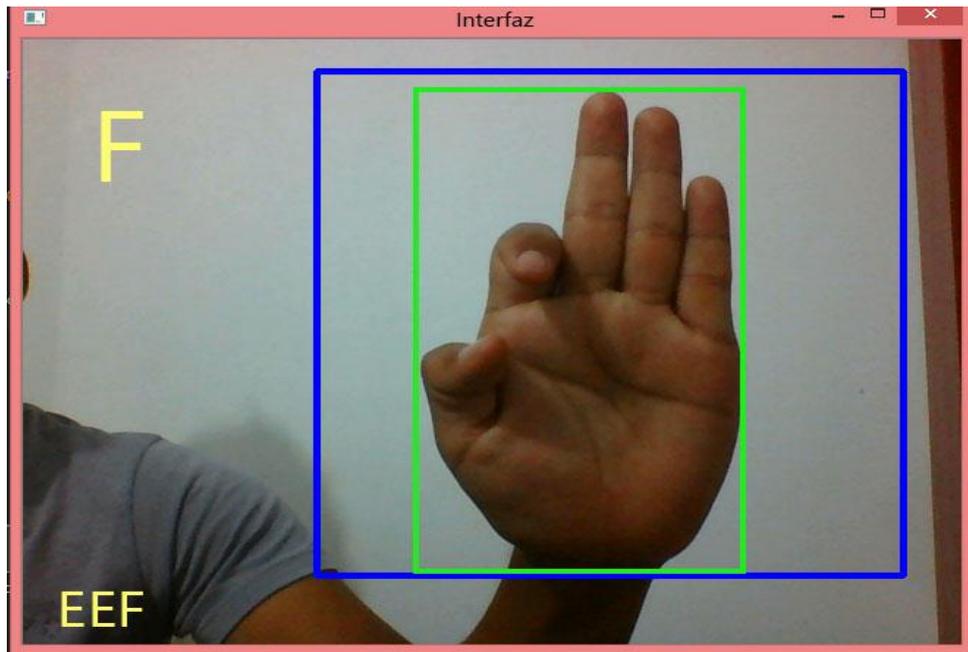
El diseño de la interfaz del modelo se centró básicamente en una sola ventana donde se ejecuta automáticamente la cámara del equipo, dentro de esta ventana se ubicó un cuadrado de 400x400 pixeles, con bordes de color azul. A su vez, este cuadrado contiene otro cuadrado con bordes de color verde que rodea solo el objeto de interés, es decir, la mano.

Mientras el modelo se encuentra en ejecución y dentro del cuadrado verde se detecta la forma de la mano, en la esquina superior izquierda se va mostrando la letra que se traduce de la posición de la mano y en la esquina inferior izquierda se van mostrando las letras concatenadas que el modelo va reconociendo como posibilidades del resultado final de la traducción.

A continuación, se muestra un mockup representativo y la interfaz real.



*Ilustración N° 61: Mockup de la interfaz del modelo*



*Ilustración N° 62: Interfaz real de modelo de reconocimiento*

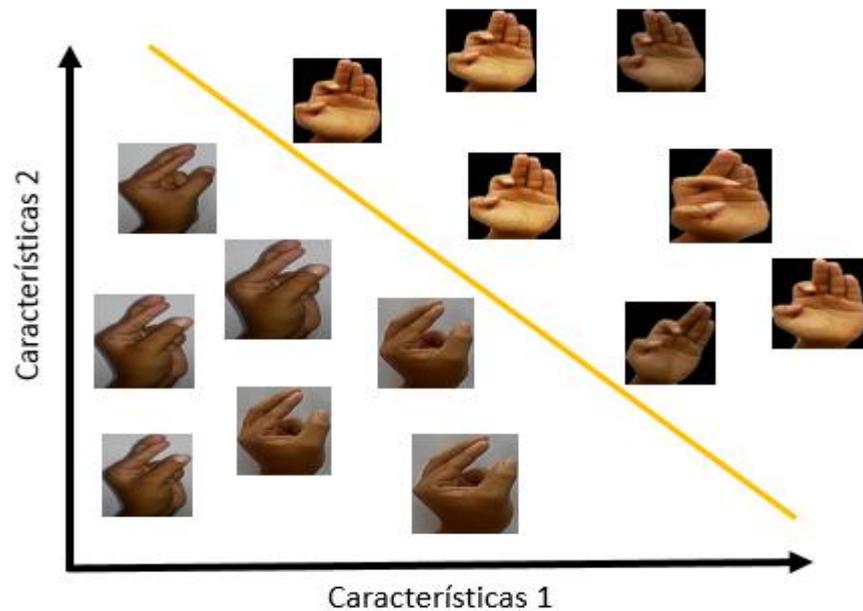
Como se puede apreciar en el diseño de la interfaz, se delimita la detección de la mano en una ubicación específica donde la cámara procesa la forma de la mano que se está capturando, esto se debe a que los objetivos de la investigación no están orientados a detectar donde está ubicada la mano, lo que se busca es la traducción de las señas que son representadas con la mano.

#### **4.6.4. Clasificadores**

Para el proceso de clasificación se hizo uso de los SVM (Support Vector Machines), ya que son considerados modelos discriminativos así como la regresión logística y redes neuronales, estos modelos se caracterizan por hacer uso de un conjunto de entrenamientos para el proceso de clasificación.

Los SVM son sistemas de clasificación binarios, es decir, permiten distinguir entre dos clases de modelos, su característica principal es la clasificación lineal, donde la solución consiste en encontrar el margen máximo entre las dos clases a partir de los vectores de soporte.

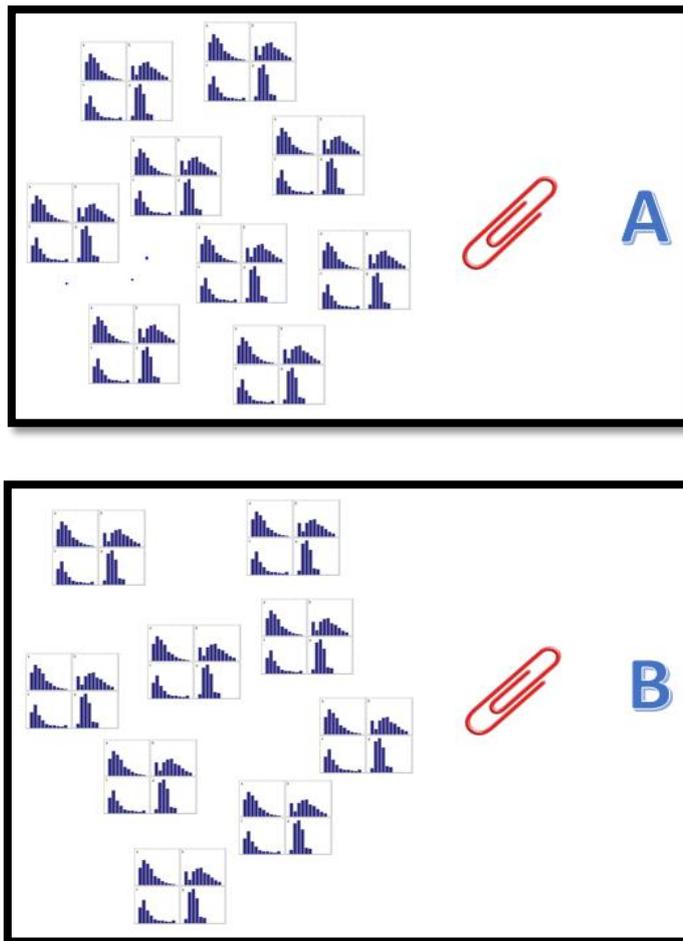
A continuación, se muestra un ejemplo del clasificador SVM, donde la línea naranja que atraviesa la imagen en forma diagonal, representa el hiperplano como frontera de decisión del modelo.



*Ilustración N° 63: Ejemplo ilustrativo del Clasificador SVM*

Es importante resaltar que, a diferencia de los modelos generativos, un SVM no busca calcular la densidad de probabilidad de cada clase, por el contrario, busca la frontera de separación a partir de las muestras de entrenamiento.

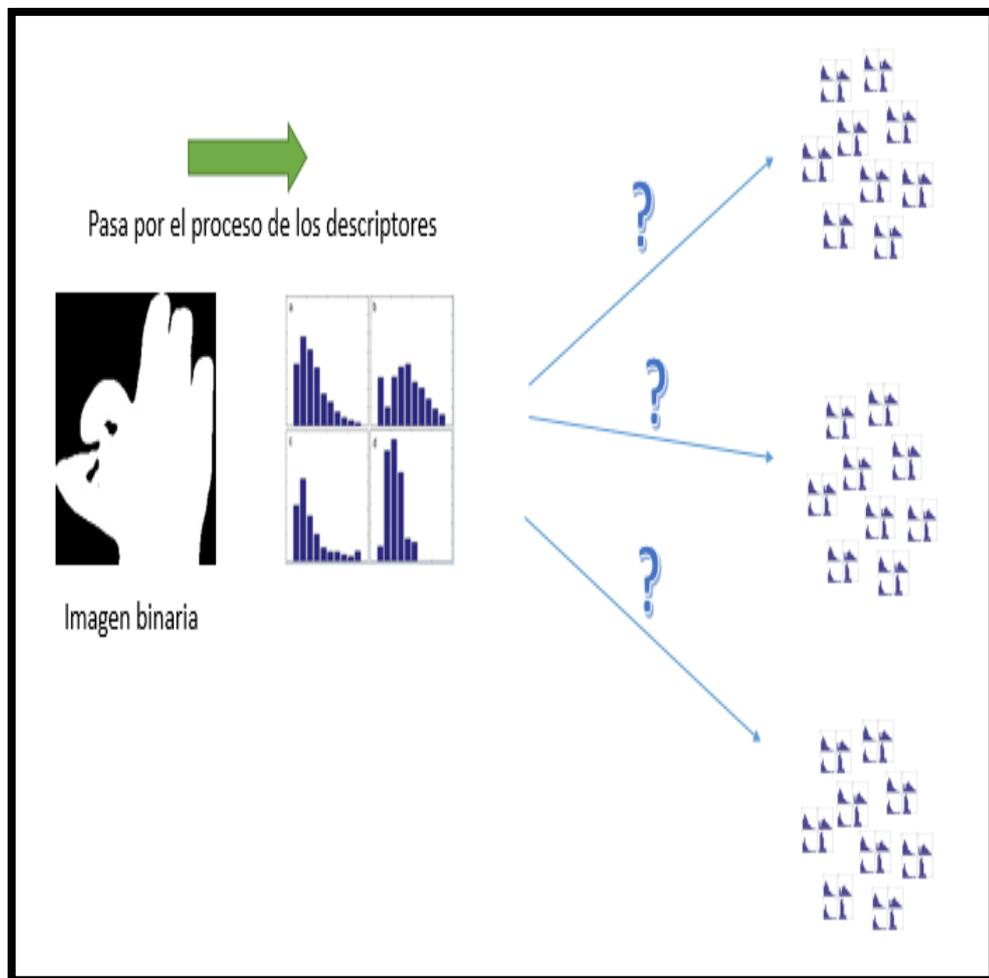
Después de la obtención de los histogramas de cada imagen, se realiza el proceso de clasificación a través de los SVM, para ello es necesario etiquetar con la letra que le corresponde a cada grupo resultante de la clasificación, este etiquetado se da conforme se va realizando el proceso de clasificación. A nivel de programación, el proceso es mucho más fácil y eficaz, puesto que cada una de las fotos recolectadas para la fase de entrenamiento lleva por nombre la letra que es representada en la imagen, por ejemplo “G-1”, gracias a estos nombres asignados, es posible agrupar y clasificar las imágenes.



*Ilustración N° 64: Etiquetado de imágenes con las letras correspondientes*

#### 4.6.5. Seleccionador

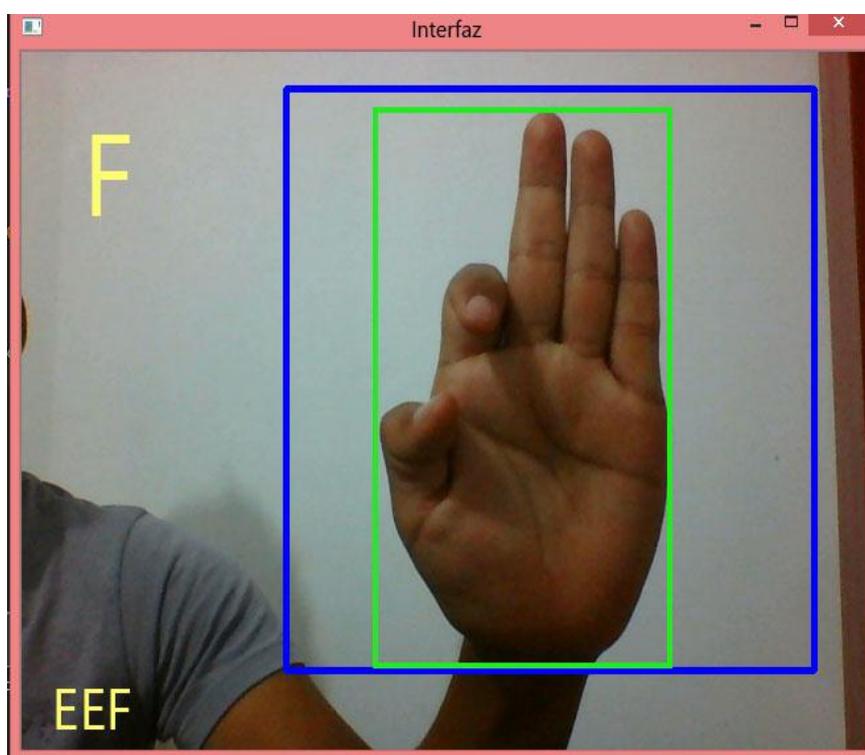
El seleccionador tiene como elemento esencial la imagen binaria que se obtuvo en el procesamiento de imágenes, esta imagen cumple la función de seleccionador y se ubica en uno de los grupos ya clasificados anteriormente. Esta imagen binaria también debe pasar por el proceso de descriptores con la finalidad de extraer sus características y poder ubicarlo en un grupo clasificado.



*Ilustración N° 65: Proceso de selección*

#### 4.6.6. Pruebas de la traducción

En esta etapa se realizan las pruebas de rendimiento de la traducción del lenguaje de señas y la corrección de algunas incidencias que se van presentando en la última fase del desarrollo del modelo. Además, se toma en cuenta los ambientes (Iluminación, fondo, etc) con los que se obtiene mejor rendimiento del sistema.



*Ilustración N° 66: Pruebas de reconocimiento*

En este punto, se presenta una infografía del proceso completo que realiza el modelo de Visión Computacional para la traducción del lenguaje de señas a texto en idioma español.

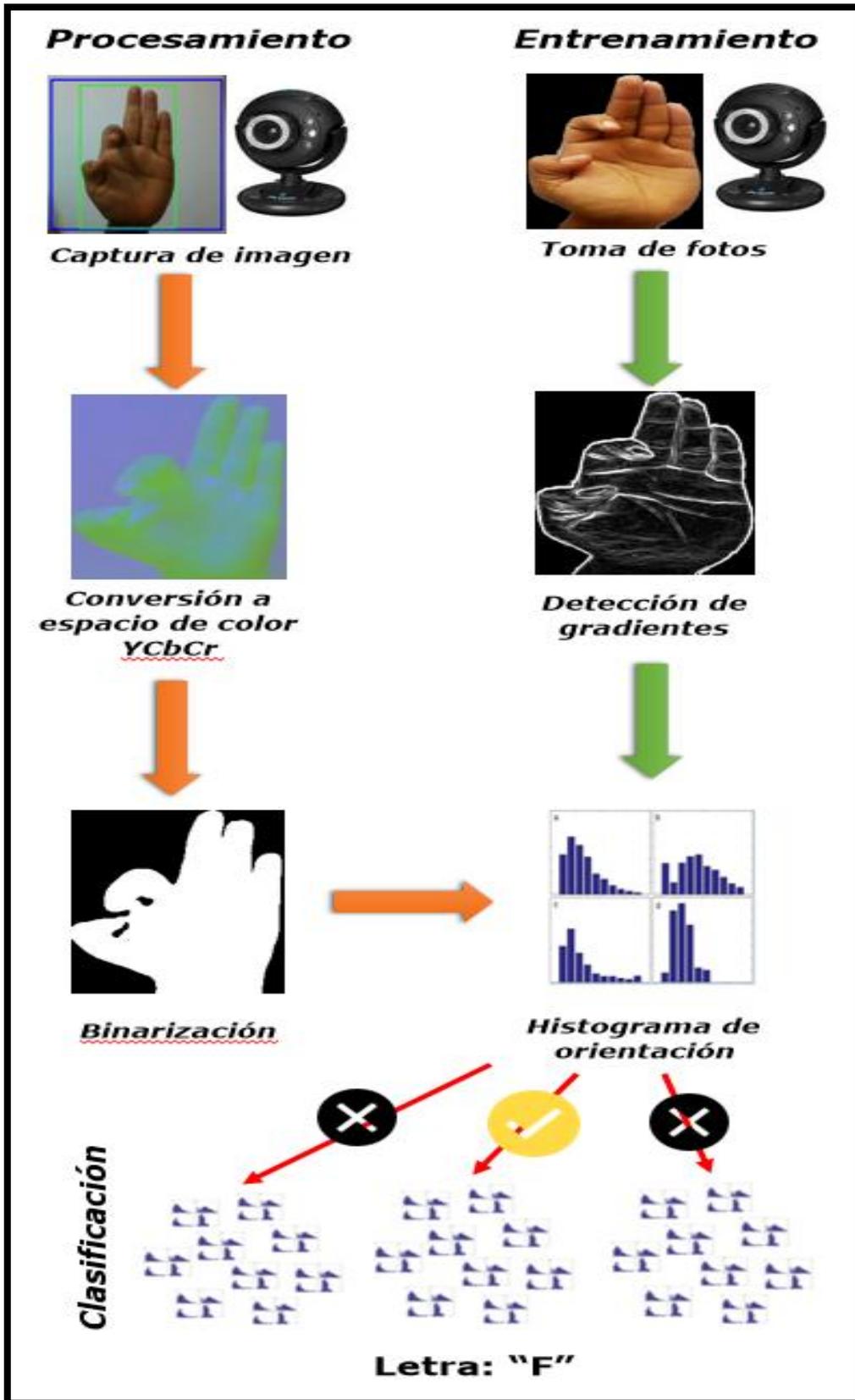


Ilustración N° 67: Infografía del Modelo de Visión Computacional

## CAPÍTULO V: MATERIALES Y MÉTODOS

### 5.1. Método de Investigación

#### 5.1.1. Propuesta experimental

La propuesta de estudio de la investigación es el diseño con pos prueba a través de un grupo de control. Para este diseño se eligen dos grupos:



Dónde:

**GE:** Grupo Experimental

**O<sub>1</sub>:** Pre prueba (Traducción del lenguaje de señas a texto en idioma español antes de la propuesta) al grupo experimental.

**X:** Visión computacional

**O<sub>2</sub>:** Pos prueba (Traducción del lenguaje de señas a texto en idioma español después de la propuesta) al grupo experimental.

#### 5.1.2. Población

Totalidad del fenómeno a estudiar en donde las unidades de población poseen una característica común, la cual se estudia y da origen a los datos de la investigación (Tamayo & Tamayo, 2002).

En el caso de esta investigación, el universo objeto de estudio, es una población finita, que está constituida por todos los estudiantes con limitaciones auditivas y del habla, siendo un total de 10 alumnos matriculados en los diferentes grados del Centro de Educación Básica Especial N° 01 de la ciudad de Chimbote.

### **5.1.3. Muestra**

Teniendo en cuenta que la población está conformada por un número pequeño de estudiantes y que se requiere de un grupo experimental que utilice correctamente el lenguaje de señas o al menos tenga un manejo básico de este medio de comunicación, la muestra está constituida por el total de la población de esta investigación.

### **5.1.4. Técnica e Instrumentos de Recolección de Datos**

#### **5.1.4.1. Técnicas**

##### **✓ De Campo**

Para determinar el nivel de impacto de la Visión Computacional, en las unidades de la muestra, en la percepción posterior.

##### **✓ De Gabinete**

Para hacer el análisis y evaluación homogénea de las unidades de la población y determinar las unidades de la muestra y sus correspondientes unidades de análisis, que conformarán el grupo único.

#### **5.1.4.2. Instrumentos**

##### **✓ Tablas de Referencia**

Preparación de los cuadros para recoger la información en relación al impacto de la Visión Computacional en las unidades de análisis en la observación posterior. Preparación de tablas para consolidar la información de los procesos antes mencionados.

##### **✓ Cuestionarios**

Para registrar información requerida, impresiones y sugerencias en cuanto al impacto de la Visión Computacional en las unidades de análisis de la muestra en función de los objetivos.

##### **✓ Tablas de Calificación**

Para recoger la información sobre el nivel de rapidez y precisión de la traducción lograda a través de las unidades de análisis de la muestra.

#### **5.1.5. Metodología de Trabajo**

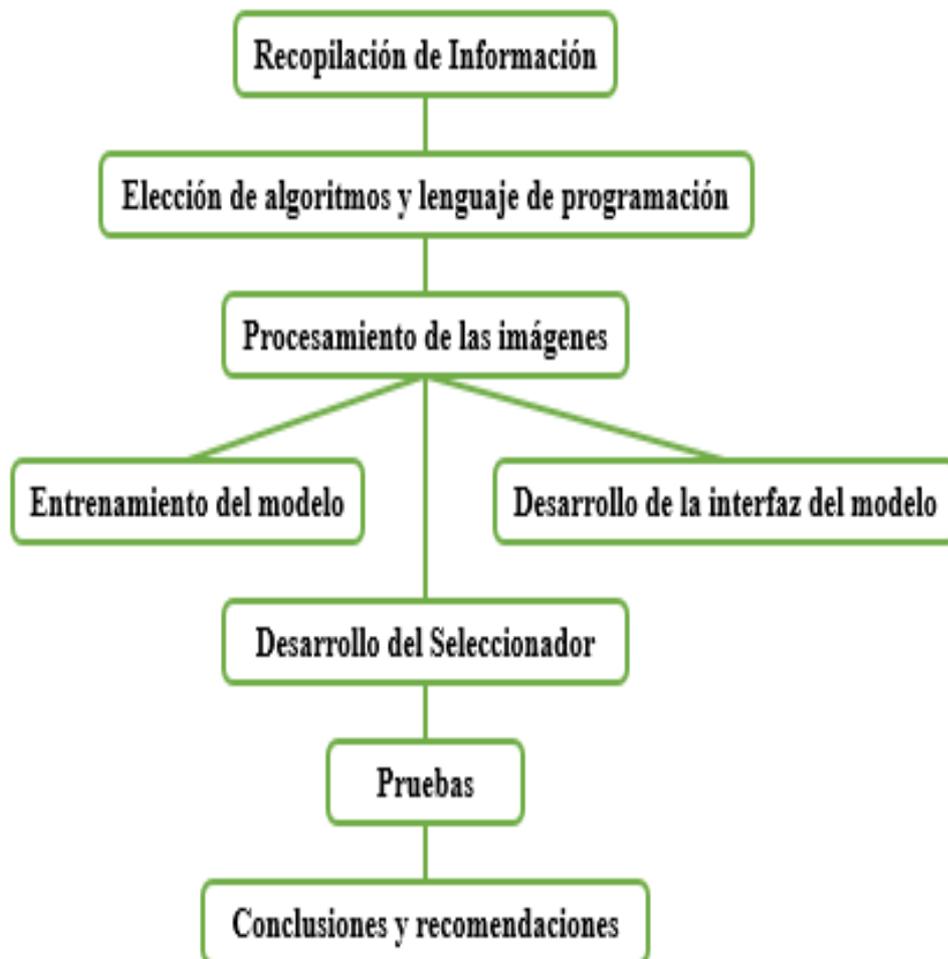
- El método de investigación a utilizar es el Inductivo – Deductivo. Ante esta realidad observable, la variable dependiente se puede dividir en características o indicadores de cada uno de los procesos definidos como: aporte de la investigación en la mejora de la comunicación entre las personas con limitaciones auditivas y del

habla y su entorno social, impacto del uso de la Visión Computacional en la traducción del Lenguaje de Señas, rapidez y precisión en la traducción del Lenguaje de Señas al relacionarla en término de causa efecto con la variable independiente, a efectos de contrastar la hipótesis, que será verdadera a través de la veracidad de los indicadores, es decir bastará con que los indicadores de la variable dependiente sean verdaderos para que la hipótesis sea verdadera, cuyos resultados permitirán generalizar a la población.

- Comparación de los resultados obtenidos en la medición del tiempo de traducción del Lenguaje de Señas a texto en idioma español haciendo uso de la Visión Computacional, y el tiempo que le lleva a una persona con conocimiento básico del Lenguaje de Señas, interpretar el mensaje que le transmite una persona con limitaciones auditivas o del habla a través del Lenguaje de Señas.
  
- Elaboración definitiva del Marco Teórico.
  
- Identificar las unidades de la población del estudio y determinar las unidades de la muestra.
  
- Preparación de las Técnicas, Instrumentos y Herramientas a utilizar en el estudio para la recogida de datos.

- Desarrollo de la investigación de la Visión Computacional, en función a la variable independiente, que permitan llevar a cabo la evaluación de los indicadores de la variable dependiente.
- Aplicar Metodologías que permitan contrastar la hipótesis a la realidad problemática.
- Elaboración del informe final de la investigación

#### 5.1.6. Planeación de la investigación



*Ilustración N° 68: Planeación de la Investigación.*

## CAPÍTULO VI: RESULTADOS Y DISCUSIONES

A continuación, se presentan los principales resultados de la investigación “Visión Computacional para la traducción en tiempo real del Lenguaje de Señas a texto en idioma español”:

- En la actualidad, el procesamiento de imágenes ha tenido un gran impacto en diferentes ambientes de investigación, puesto que, por medio de la captura de imágenes se obtiene información detallada para llevar a cabo diversos análisis que contribuyen a mejorar las distintas necesidades del ser humano. El presente trabajo tiene un gran impacto social, ya que en la ciudad de Chimbote no es común el desarrollo de investigaciones tecnológicas orientadas a mejorar la calidad de vida de personas con limitaciones auditivas o del habla, es por ello que el CEBE N° 1, institución que brinda una educación especializada para este sector de la población, se encuentra interesado en que el resto de los ciudadanos tengan conocimiento del lenguaje de señas con el fin de disminuir las brechas de discriminación que existen en la sociedad.
- El modelo desarrollado en esta investigación se basa en la traducción del lenguaje de señas con la finalidad de que una persona pueda interpretar correctamente el mensaje que quiere transmitir otra persona que tiene limitaciones auditivas o del habla y que utiliza las señas como medio de comunicación. Puesto que en la actualidad las personas que tienen este tipo de limitaciones se enfrentan a barreras sociales que no les permiten

integrarse en la comunidad donde viven y contar con las mismas oportunidades laborales que tienen las demás personas, esta investigación utilizó la Visión Computacional y logró la traducción en tiempo real del Lenguaje de Señas a texto en idioma español.

- Para el desarrollo del modelo, en un inicio se empleó el lenguaje de programación Java, sin embargo, conforme se especificó en capítulos anteriores de este informe, se optó por el lenguaje de programación Python ya que se contaba con mayor documentación sobre la fase de entrenamiento y la aplicación de los SVM.
- Para el procesamiento de imágenes se empleó un espacio de color alternativo al BGR, esto debido a los diferentes tonos de piel de las manos y la variación de iluminación en los ambientes, por lo tanto, se empleó el espacio de color YCbCr, donde se toma en cuenta la luminosidad y los tonos de piel, logrando una segmentación más exacta de la forma de la mano.
- Para comprobar la precisión y rapidez en la traducción del lenguaje de señas a través del modelo de Visión Computacional (V.C.) se llevó a cabo dos pruebas donde participaron los 10 estudiantes con limitaciones auditivas o del habla (sordomudos) que conforman en su totalidad la muestra de esta investigación:
  - Para el desarrollo de la Prueba A (pre prueba) se requirió de la colaboración de 10 personas adicionales a la muestra, con un conocimiento mínimo del lenguaje de señas y sin ninguna limitación

auditiva. El grupo de personas que no tenían limitaciones auditivas debían traducir las letras que eran representadas por los estudiantes sordomudos a través del lenguaje de señas.

- Para el desarrollo de la Prueba B (post prueba), los estudiantes sordomudos hicieron uso del modelo de Visión Computacional (V.C.) para la traducción del lenguaje de señas.

- En las pruebas realizadas, los estudiantes del CEBE N° 1 utilizaron la mano izquierda para representar las letras del alfabeto dactilológico peruano, además este grupo de estudiantes estaba conformado por varones y mujeres con diferentes tonos de piel, obteniéndose los siguientes resultados:

**a. En relación a la precisión en la traducción del lenguaje de señas:**

Uno de las dificultades en la traducción del lenguaje de señas, es la similitud que existe en la representación de algunas letras, como es el caso de la letra “A” y la letra “S”, donde la posición que adquiere la mano cuando representa una de estas letras es muy parecida, esto ocasiona errores en la traducción y por consiguiente, la comunicación entre personas con un conocimiento mínimo del lenguaje de señas y una personas sordomuda se hace cada vez más difícil.

Para evaluar estos casos y comprobar el cumplimiento del objetivo específico de optimizar la precisión en la traducción del lenguaje de señas a texto en idioma español con el uso de la Visión Computacional, se comparó el porcentaje de aciertos obtenidos en la

Prueba A (pre prueba) y la Prueba B (post prueba), mostrando los siguientes resultados:

*Tabla N° 8: Porcentaje de aciertos en el proceso de traducción*

N°	Imagen	Letra	% Aciertos	
			Prueba A (Sin el modelo de V.C.)	Prueba B (Con el modelo de V.C.)
1		A	80%	70%
2		B	40%	80%
3		C	100%	100%
4		D	20%	80%
5		E	30%	50%
6		F	0%	60%
7		G	10%	60%

8		H	0%	50%
9		I	60%	80%
10		K	20%	60%
11		L	80%	90%
12		M	50%	40%
13		N	30%	30%
14		O	100%	90%
15		P	60%	70%
16		Q	0%	70%
17		R	10%	70%

18		S	0%	50%
19		T	40%	70%
20		U	60%	90%
21		V	80%	70%
22		W	70%	70%
23		X	0%	90%
24		Y	10%	80%
<b>Promedio (<math>\bar{X}_A, \bar{X}_B</math>)</b>			<b>40%</b>	<b>70%</b>
<b>Desviación Estándar</b>			<b>0.3355452</b>	<b>0.17315278</b>
<b>Mayor % de acierto</b>			<b>100%</b>	<b>100%</b>
<b>Menor % de acierto</b>			<b>0%</b>	<b>30%</b>
<b>Varianza Muestral (<math>S_A, S_B</math>)</b>			<b>0.11259058</b>	<b>0.02998188</b>

Como se puede apreciar en la tabla de resultados, en la prueba A, donde no se utilizó el modelo de V.C., se obtuvo un promedio de

40% de aciertos, mientras que en la prueba B, con el uso del modelo de V.C., se obtuvo un promedio de 70% de aciertos; por lo tanto, se puede concluir que el modelo de Visión Computacional desarrollado en esta investigación, contribuye significativamente en la precisión de la traducción del lenguaje de señas a texto en idioma español.

**Hipótesis Nula.** No existen mejoras significativas en los porcentajes de aciertos obtenidos en las pruebas A y B.

$$H_0: \mu_A \geq \mu_B$$

**Hipótesis Alternativa:** Existen mejoras significativas en los porcentajes de aciertos obtenidos en las pruebas A y B.

$$H_a: \mu_A < \mu_B$$

▪ **Prueba de comparación de varianzas:**

*Tabla N° 9: Comparación de Varianzas de los aciertos*

Comparación de Varianzas					
$(S_A)^2$	$(S_B)^2$	Valor Experimental	Nivel de Significancia	T-student T(0.01,10-1)	Distribución F. F(0.01,10-1,10-1)
0,01267664	0,00089891	14,1021805	$\alpha = 0.01$	2,8214	5.351
<b>Contraste:</b>	5.351 < 14,1021805		<b>Resultado:</b>	<i>Si es Significativa</i>	

La prueba de comparación de varianzas según los datos de la tabla resulta significativa con un margen de error del 1% lo que permite concluir que existen mejoras significativas en los

porcentajes de aciertos entre las varianzas de ambos grupos. Esto permite determinar que se acepta la Hipótesis Alternativa.

▪ **Prueba de comparación de medias:**

Nivel de Significancia:  $\alpha = 0.01$   $\alpha/2 = 0.005$   $\alpha = 0.0025$

$$F(Z) = 0.99 + 0.0025 = 0.9925$$

Valor en la tabla de Distribución Normal:

$$Z = 2.43$$

- Cálculo de los valores experimentales:

$$Z_0 = \frac{\bar{X}_{1GA} + \bar{X}_{1GB}}{\sqrt{\frac{(S_{1GA})^2}{n_1} + \frac{(S_{1GB})^2}{n_2}}} \quad Z_0 = 29,62861158$$

**R.A.:** Región de Aceptación.

**R.R.:** Región de Rechazo.

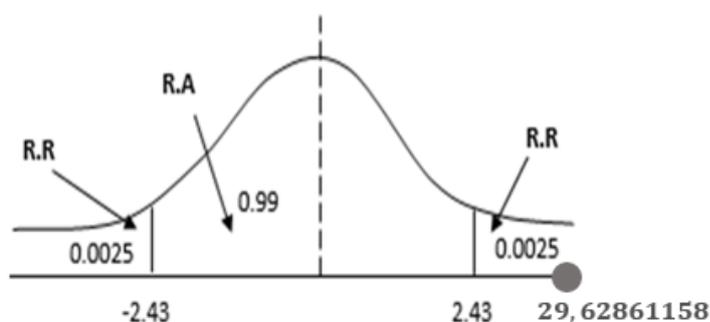


Tabla N° 10: Comparación de Medias de los aciertos

Comparación de Medias					
$(S_A)^2$	$(S_B)^2$	Valor Experimental	Nivel de Significancia	T-student T(0.01,10-1)	Distribución F. F(0.01,10-1,10-1)
0,01267664	0,00089891	29,62861158	$\alpha = 0.01$	2,8214	5.351
<b>Resultado:</b>		<i>Si es Significativa</i>			

Según el resultado de la prueba se observa que el resultado promedio de la prueba B es mayor que el resultado promedio de la prueba A, esto demuestra que existen mejoras significativas en los porcentajes de aciertos a un nivel del 1% de error, por lo tanto se acepta la Hipótesis Alternativa.

**b. En relación a la rapidez en la traducción del lenguaje de señas:**

Otra de las desventajas en la comunicación con personas sordomudas, es la rapidez con que se logra traducir un mensaje, si bien hay letras que son fáciles de reconocer o que son más conocidas por las demás personas, también hay letras del abecedario dactilológico que tienen cierto grado de dificultad en la posición de la mano y muchas veces las personas demoran en reconocerlas.

Para evaluar estas situaciones y comprobar el cumplimiento del objetivo específico de mejorar el tiempo empleado en la traducción de una determinada letra del lenguaje de señas a través del modelo de Visión Computacional, se comparó los tiempos obtenidos en la Prueba A (pre prueba) y la Prueba B (post prueba), mostrando los siguientes resultados:

Tabla N° 11: Tiempo en cada prueba del proceso de traducción

N°	Imagen	Letra	Tiempo (Segundos)	
			Prueba A (Sin el modelo de V.C.)	Prueba B (Con el modelo de V.C.)
1		A	5.7	4.5
2		B	11.4	5
3		C	3.3	4.8
4		D	12.5	4.3
5		E	12	4.6
6		F	13	4.6
7		G	12.3	4.4
8		H	12	5

9		I	7.8	4.5
10		K	13	4.1
11		L	3.7	4.2
12		M	8.1	4.6
13		N	10.1	5.2
14		O	2.7	3.9
15		P	8.9	4.4
16		Q	13.7	4.2
17		R	14	4
18		S	13.8	4.1

19		T	6.2	4.6
20		U	5.4	4.3
21		V	6.6	4
22		W	6.9	3.5
23		X	13	4.2
24		Y	12.1	4.3
<b>Promedio</b>			<b>9.50833333</b>	<b>4.3875</b>
<b>Desviación Estándar</b>			<b>3.69511036</b>	<b>0.3848348</b>
<b>Mayor tiempo</b>			<b>14</b>	<b>5.2</b>
<b>Menor tiempo</b>			<b>2.7</b>	<b>3.5</b>
<b>Varianza Muestral</b>			<b>13.6538406</b>	<b>0.14809783</b>

Como se puede apreciar en la tabla de resultados, en la prueba A, donde no se utilizó el modelo de V.C., se obtuvo un promedio de 9.5 segundos, mientras que en la prueba B, con el uso del modelo de V.C., se obtuvo un promedio de 4.3 segundos; por lo tanto, se puede

concluir que el modelo de Visión Computacional desarrollado en esta investigación, contribuye significativamente en la rapidez de la traducción del lenguaje de señas a texto en idioma español.

**Hipótesis Nula.** No existen mejoras significativas en los tiempos de traducción obtenidos en las pruebas A y B.

$$H_0: \mu_A = \mu_B$$

**Hipótesis Alternativa:** Existen mejoras significativas en los tiempos de traducción obtenidos en las pruebas A y B.

$$H_a: \mu_A < \mu_B$$

- **Prueba de comparación de varianzas:**

*Tabla N° 12: Comparación de Varianzas del tiempo de traducción*

Comparación de Varianzas					
$(S_A)^2$	$(S_B)^2$	Valor Experimental	Nivel de Significancia	T-student T(0.01,10-1)	Distribución F. F(0.01,10-1,10-1)
68,0191645	0,17252586	394,254903	$\alpha = 0.01$	2,8214	5.351
<b>Contraste:</b>	5.351 < 394,254903		<b>Resultado:</b>	<i>Si es Significativa</i>	

La prueba de comparación de varianzas según los datos de la tabla resulta significativa con un margen de error del 1% lo que permite concluir que existen mejoras significativas en los tiempos de traducción entre las varianzas de ambos grupos. Esto permite determinar que se acepta la Hipótesis Alternativa.

- **Prueba de comparación de medias:**

Nivel de Significancia:  $\alpha = 0.01$   $\alpha/2 = 0.005$   $\alpha = 0.0025$

$$F(Z) = 0.99 + 0.0025 = 0.9925$$

Valor en la tabla de Distribución Normal:

$$Z = 2.43$$

- Cálculo de los valores experimentales:

$$Z_0 = \frac{\bar{X}_{1GA} + \bar{X}_{1GB}}{\sqrt{\frac{(S_{1GA})^2}{n_1} + \frac{(S_{1GB})^2}{n_2}}} \quad Z_0 = 5,050062402$$

**R.A.:** Región de Aceptación.

**R.R.:** Región de Rechazo.

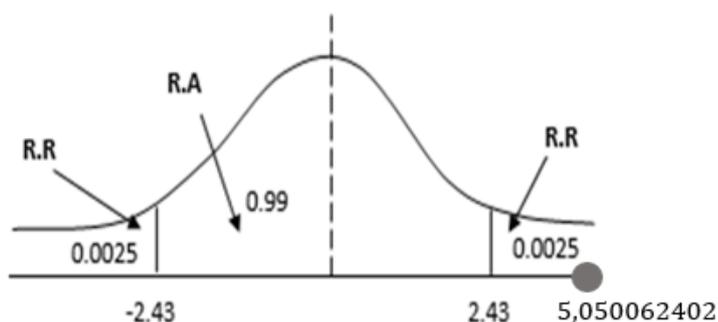


Tabla N° 13: Comparación de Medias del tiempo de traducción

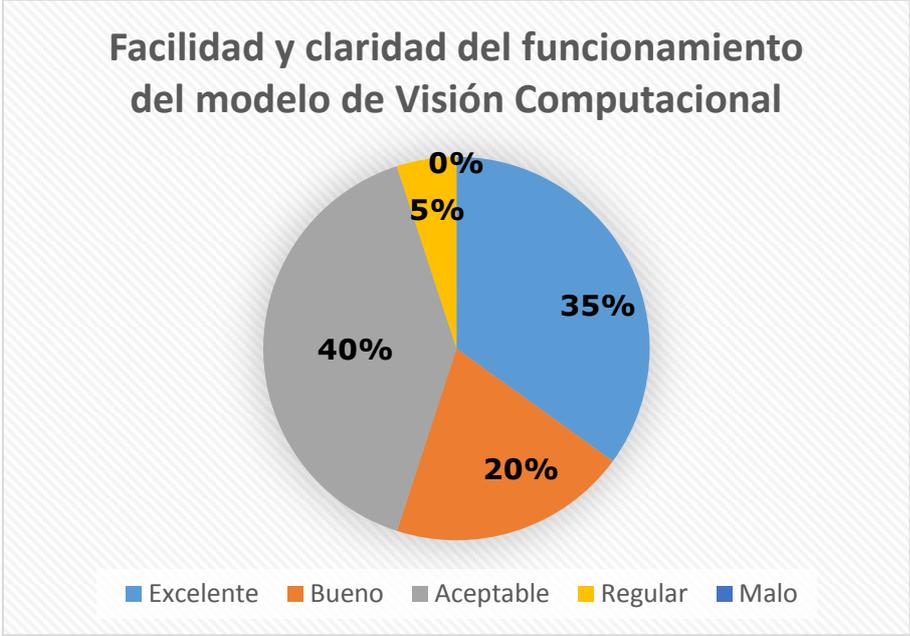
Comparación de Medias					
$(S_A)^2$	$(S_B)^2$	Valor Experimental	Nivel de Significancia	T-student T(0.01,10-1)	Distribución F. F(0.01,10-1,10-1)
0,01267664	0,00089891	5,050062402	$\alpha = 0.01$	2,8214	5.351
<b>Resultado:</b>		<i>Si es Significativa</i>			

Según el resultado de la prueba se observa que el resultado promedio de la prueba B es menor que el resultado promedio de

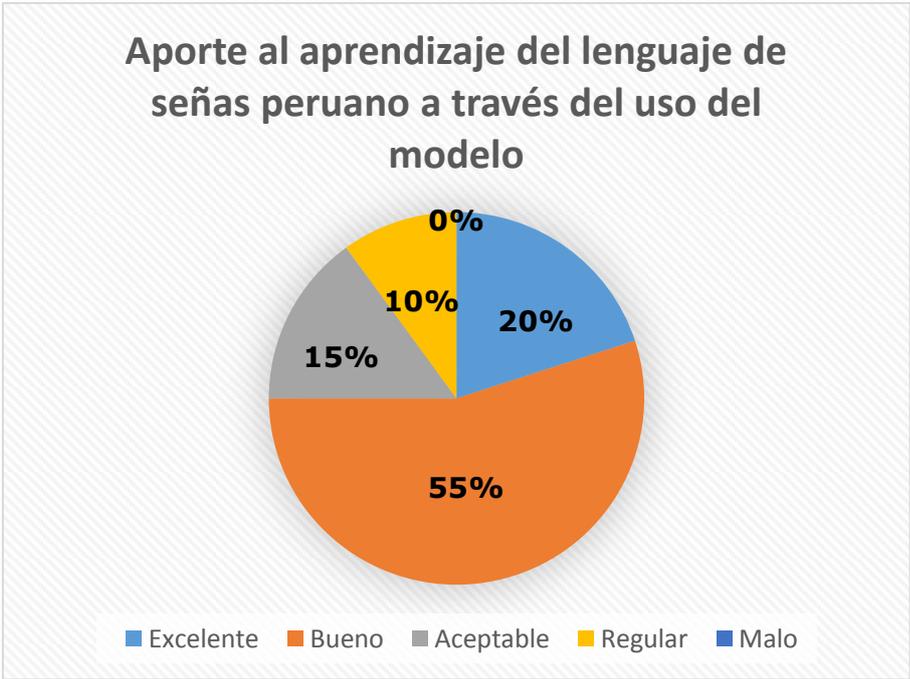
la prueba A, esto demuestra que existen mejoras significativas en los tiempos de traducción a un nivel del 1% de error, por lo tanto se acepta la Hipótesis Alternativa.

- c. En relación a la usabilidad del modelo:** De igual forma, es importante saber el impacto que tuvo el desarrollo de esta investigación, para ello se evaluó la opinión de las personas que participaron en las pruebas de la investigación, respecto al uso del modelo de Visión Computacional como medio para la traducción del lenguaje de señas.

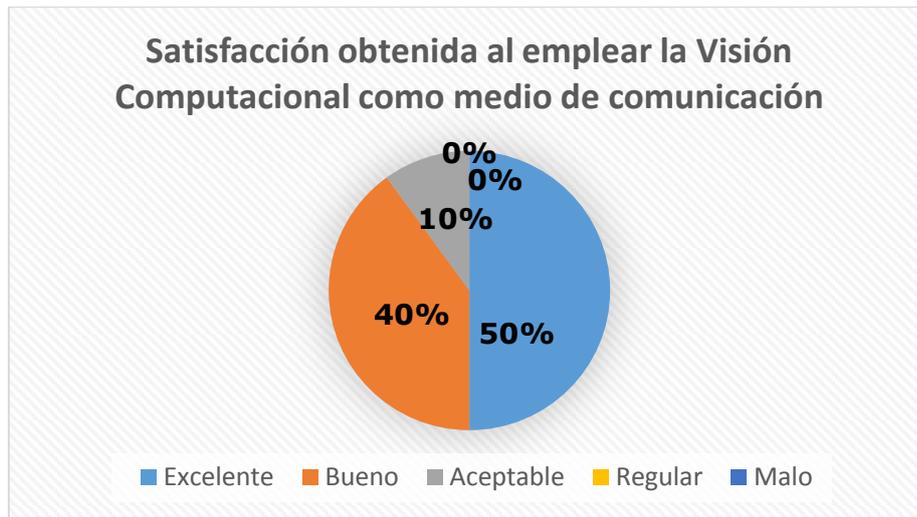
Para comprobar el objetivo específico sobre la usabilidad que tiene el modelo de Visión Computacional como herramienta tecnológica para la traducción del lenguaje de señas, se encuestó a todas las personas que participaron de las pruebas de precisión y rapidez del modelo, siendo un total de 20 personas que respondieron al cuestionario sobre el funcionamiento del modelo, el grado de aporte en el aprendizaje del lenguaje de señas y el nivel de satisfacción al usar esta tecnología, obteniendo los siguientes resultados:



*Ilustración N° 69: Resultados de la Pregunta N°1 de la encuesta*



*Ilustración N° 70: Resultados de la Pregunta N°2 de la encuesta*



*Ilustración N° 71: Resultados de la Pregunta N°3 de la encuesta*

Para validar el instrumento de recolección de datos se aplicó el alfa de Cronbach (ver anexos) obteniendo un coeficiente de 0.86, lo que significa una fiabilidad respetable, con mediciones estables y consistentes.

Como se observa en la gráfica de barras existe un alto número de estudiantes que se encuentran más satisfechos luego de utilizar el modelo de Visión Computacional para la traducción del lenguaje de señas, puesto que pueden comunicarse de forma más rápida y precisa con las personas que los rodean. Por lo tanto, en base a los resultados de esta evaluación, se puede concluir que el modelo de Visión Computacional tiene la aceptación de los estudiantes.

En base a lo antes expuesto se demuestra que la hipótesis: “La visión computacional permite traducir en tiempo real el lenguaje de señas a texto en idioma español”, es verdadera en relación al análisis de los indicadores antes explicados.

## CONCLUSIONES

En el desarrollo de esta investigación se concluye que mediante el uso de la Visión Computacional se logró la traducción en tiempo real del Lenguaje de Señas a texto en idioma español.

En conclusiones específicas se menciona lo siguiente:

- Con base en los resultados obtenidos en la evaluación de precisión en el proceso de traducción, se concluye que el modelo de Visión Computacional optimizó la traducción del lenguaje de señas a texto en idioma español con un nivel de precisión del 70% de aciertos frente al 40% de aciertos que se obtuvo sin la aplicación del modelo, estos resultados también permiten comprobar que con el uso del modelo de Visión Computacional, una persona con conocimientos mínimos de este lenguaje puede interpretar con precisión las principales señas del alfabeto dactilológico peruano.
- De acuerdo con los resultados obtenidos en la evaluación de rapidez en el proceso de traducción, se concluye que el modelo de Visión Computacional mejoró el tiempo de traducción de una determinada letra del lenguaje de señas a texto en idioma español, obteniendo un tiempo promedio de 4.4 segundos en comparación al tiempo promedio de 9.5 segundos que se obtuvo sin el uso de esta herramienta tecnológica; estos resultados a su vez, permiten comprobar que el uso de la Visión Computacional puede mejorar la comunicación entre personas sordomudas y su entorno social.

- Considerando los resultados obtenidos en el análisis de precisión de la traducción del lenguaje de señas, se establece una mejora considerable en el porcentaje de aciertos; sin embargo, algunas letras como la “A”, “O”, “V” tienen un porcentaje menor de aciertos con el modelo de visión computacional, debido a que existen otras letras donde las regiones segmentadas tienen formas similares.
- El algoritmo de HOG permitió de manera satisfactoria la extracción de características de una imagen, proceso fundamental para la clasificación de las imágenes en la fase de entrenamiento del modelo, detallado en el Capítulo V, y la selección del resultado final para la imagen procesada.
- Se recolectó un conjunto de imágenes con las cuales se formó una base de datos de las posiciones de la mano que representan las letras del alfabeto dactilológico peruano, estas posiciones solo emplean una mano y no requieren de un movimiento adicional para la formación de la letra.
- Esta investigación logró facilitar la comunicación entre personas con limitaciones auditivas o del habla y su entorno, ya que se empleó la visión Computacional como un medio de traducción y aprendizaje del lenguaje de señas.

- Se realizó pruebas del modelo que permitieron comprobar su correcto funcionamiento en cuanto a la precisión y la rapidez de la traducción de las letras del alfabeto dactilológico peruano.
  
- El desarrollo de esta investigación brindó una perspectiva más completa sobre las diversas dificultades que atraviesan las personas sordomudas al momento de comunicarse con las demás personas de su entorno, además se presenció la falta de tolerancia que existe frente a las limitaciones físicas y psicológicas que atraviesan muchos niños y jóvenes en la actualidad.
  
- El CEBE N° 1 de la ciudad de Chimbote otorgó las facilidades para la culminación exitosa de este proyecto, permitiendo conocer la ardua labor que realiza esta institución, con la finalidad de otorgar una mejor calidad de vida a las personas con limitaciones auditivas o del habla.
  
- El estudio realizado brinda las bases para futuras investigaciones donde se proponga un mayor desarrollo del campo de la tecnología de la visión computacional, buscando siempre que estos aportes logren reducir las brechas de la discriminación que existe en la sociedad.

## RECOMENDACIONES

- Se recomienda trabajar en un ambiente controlado para obtener mejores resultados si las imágenes a procesar se transforman a escala de grises, puesto que la variación de luminosidad en el fondo o las diferentes tonalidades de la piel pueden afectar la precisión en la segmentación de las imágenes.
- Se sugiere que las imágenes empleadas para el proceso de entrenamiento del modelo no sean pesadas, ya que se puede elegir una mayor cantidad de imágenes, dependiendo también de la capacidad computacional del equipo donde sea ejecutado el modelo.
- Si hace uso de Máquinas de Vectores de Soporte (SVM), es recomendable trabajar con el lenguaje de programación Python, debido a que cuenta mayor documentación sobre la aplicación de este modelo de aprendizaje supervisado.
- Se recomienda trabajar con la versión 2.7 de Python por ser la más estable hasta la fecha de finalización de esta investigación, ya que permite una mejor integración con las librerías de OpenCV versión 2.4.9.
- Para futuras investigaciones, se podría considerar el procesamiento de letras que requieren más de una posición, lo que incluye un mayor número de capturas para un solo símbolo.

## REFERENCIAS BIBLIOGRÁFICAS

- Acharya , T., & K. Ray, A. (2005). *Image Processing Principles and Applications* . United States of America: A Wiley-Interscience Publication.
- Alegre Gutiérrez, E., Pajares Martinsanz, G., & De la Escalera Hueso, A. (2016). *Conceptos y Métodos en Visión por Computador*. España.
- Antón López, E. (2009). *Reconocimiento automático de lenguaje de signos:Lenguaje ASL*. Barcelona.
- C. Gonzalez, R., & E. Woods, R. (2002). *Digital Image Processing (2nd Edition)*. Prentice Hall.
- Egg, E. A. (2011). *Aprender a investigar : nociones básicas para la investigación social*. Córdoba: Brujas.
- Escalera, A. d. (2001). *Visión por Computador. Fundamentos y*. México: Prentice Hall.
- González Marcos, A., de Pisón Ascacíbar, F. J., Pernía Espinoza, A. V., Alba Elías, F., Castejón Limas, M., Ordieres Meré, J., & Vergara González, E. (2006). *Técnicas y Algoritmos Básicos de Visión Artificial*. La Rioja: Publicaciones Universidad de Rioja.
- Ministerio de Educación, Peruano. (2010). Lengua de señas peruana. En D. g. Especial. Lima.
- Mundaca Vidarte, G. A. (2016). *Detección de caracteres de placas de automóviles mediante técnicas de visión artificial*. Piura: Repositorio institucional PIRHUA – Universidad de Piura.
- Observatorio Nacional de la Discapacidad . (26 de Septiembre de 2016). <http://www.conadisperu.gob.pe>. Obtenido de <http://www.conadisperu.gob.pe/observatorio/index.php/informacion-de-base/informes/nacional/tematicos/349-informe-tematico-n-5-situacion-de-las-personas-con-discapacidad-auditiva-en-el-peru.html>
- Organización Mundial de la Salud . (2012). <http://www.who.int>. Obtenido de [http://www.who.int/pbd/deafness/WHO\\_GE\\_HL.pdf](http://www.who.int/pbd/deafness/WHO_GE_HL.pdf)
- OVIEDO, A. (1998). *Una aproximación a la Lengua de Señas Colombiana*. Bogotá: INSORMEN.
- Robles Pizarro, L. D. (2016). *Caracterización y Reconocimiento de Objetos Mediante Algoritmos de Visión Computacional para la Interacción de un Robot con su Entorno*. Lima: Tesis Pucp.
- Sampieri, R. H., Collado, C. F., & Lucio, P. B. (2010). *Metodología de la Investigación*. Mexico: McGRAW-HILL.
- Steven W, S. (1997). *The Scientist and Engineer's Guide to Digital Signal Processing*. California Technical Pub.

- Tamayo, M., & Tamayo. (2002). *El proceso de la Investigación Científica*. México: Limusa Noriega Editorial.
- Valencia Sierra, V., & Villa Brochero, B. (2014). *Diseño de un Sistema de Reconocimiento de Gestos no Móviles Mediante el Procesamiento Digital de Imágenes*. Barranquilla.
- Vélez Serrano, J. F., Moreno Díaz, A. B., Sánchez Calle, Á., & Sánchez-Marín, J. L. (2015). *Visión por computador*. Madrid.

## ANEXOS

### ➤ **Parámetros para Prueba “A”**

Grupo A: Estudiantes sordomudos

Grupo B: Persona con conocimiento mínimo del Lenguaje de Señas

Prueba "A"		
Grupos para prueba		Equipo para pruebas
Grupo "A"	Grupo "B"	
A-1	B-1	P-1
A-2	B-2	P-2
A-3	B-3	P-3
A-4	B-4	P-4
A-5	B-5	P-5
A-6	B-6	P-6
A-7	B-7	P-7
A-8	B-8	P-8
A-9	B-9	P-9
A-10	B-10	P-10

➤ **Resultados de Prueba "A"**

- Análisis de precisión

Resultados de Prueba "A"													
Nº	L/P	Nº de Prueba										Precisión	
		Letra	P-1	P-2	P-3	P-4	P-5	P-6	P-7	P-8	P-9	P-10	Nº de Aciertos
1	"A"	√	√	√	X	√	√	X	√	√	√	8	80%
2	"B"	X	√	X	√	X	X	√	√	X	X	4	40%
3	"C"	√	√	√	√	√	√	√	√	√	√	10	100%
4	"D"	X	X	√	X	X	X	X	X	√	X	2	20%
5	"E"	√	X	X	X	√	X	√	X	X	X	3	30%
6	"F"	X	X	X	X	X	X	X	X	X	X	0	0%
7	"G"	X	X	X	X	X	X	X	√	X	X	1	10%
8	"H"	X	X	X	X	X	X	X	X	X	X	0	0%
9	"I"	X	√	√	X	√	√	X	√	√	X	6	60%
10	"K"	X	√	X	X	X	X	X	X	√	X	2	20%
11	"L"	√	√	√	X	√	√	X	√	√	√	8	80%
12	"M"	X	√	X	X	√	X	√	√	√	X	5	50%
13	"N"	X	X	√	X	X	√	X	X	X	√	3	30%
14	"O"	√	√	√	√	√	√	√	√	√	√	10	100%
15	"P"	√	√	X	√	X	X	√	X	√	√	6	60%
16	"Q"	X	X	X	X	X	X	X	X	X	X	0	0%
17	"R"	X	√	X	X	X	X	X	X	X	X	1	10%
18	"S"	X	X	X	X	X	X	X	X	X	X	0	0%
19	"T"	X	√	√	X	X	X	√	X	√	X	4	40%
20	"U"	X	√	√	X	√	√	X	√	√	X	6	60%
21	"V"	X	√	√	√	√	√	X	√	√	√	8	80%
22	"W"	X	√	√	√	X	√	X	√	√	√	7	70%
23	"X"	X	X	X	X	X	X	X	X	X	X	0	0%
24	"Y"	X	X	X	X	X	X	X	X	√	X	1	10%
Total de Pruebas:		240						Total de aciertos:				95	40%

- Análisis de rapidez

Resultados de Prueba "A"												
N°	L/P	N° de Prueba										Tiempo promedio
	Letra	P-1	P-2	P-3	P-4	P-5	P-6	P-7	P-8	P-9	P-10	
1	"A"	6	5	7	3	3	3	4	6	8	4	4.9
2	"B"	11	10	13	7	9	13	9	9	12	7	10
3	"C"	4	3	4	5	4	3	3	4	3	3	3.6
4	"D"	9	10	10	8	9	11	8	9	12	8	9.4
5	"E"	9	12	12	13	9	12	9	13	10	9	10.8
6	"F"	12	9	11	9	10	12	9	11	11	10	10.4
7	"G"	11	11	12	11	12	9	11	12	12	12	11.3
8	"H"	12	12	11	10	9	11	9	13	12	12	11.1
9	"I"	6	6	5	5	8	7	8	8	8	7	6.8
10	"K"	11	11	10	8	9	8	12	12	9	7	9.7
11	"L"	5	2	2	5	3	5	2	4	2	3	3.3
12	"M"	5	8	5	8	5	10	9	7	8	10	7.5
13	"N"	8	6	8	10	9	6	8	5	6	6	7.2
14	"O"	4	2	2	4	3	4	4	4	3	4	3.4
15	"P"	12	12	7	8	11	6	9	6	11	7	8.9
16	"Q"	9	12	11	12	12	12	10	9	10	12	10.9
17	"R"	12	10	10	9	12	11	11	12	9	9	10.5
18	"S"	9	9	12	10	9	11	12	10	9	11	10.2
19	"T"	7	8	6	6	8	9	11	4	9	4	7.2
20	"U"	7	2	3	6	2	7	2	7	5	7	4.8
21	"V"	2	5	5	3	3	5	2	2	4	5	3.6
22	"W"	7	6	3	7	2	6	6	3	5	4	4.9
23	"X"	10	11	9	9	11	10	12	10	12	12	10.6
24	"Y"	12	12	12	8	12	9	8	11	9	10	10.3
Total de Pruebas:		240					Total de tiempos:					191.3

➤ **Parámetros para Prueba “B”**

Grupo A: Estudiantes sordomudos

Grupo B: Iteración con el Modelo de Visión Computacional

Prueba "B"		
Grupos para prueba		N° de pruebas
Grupo "A"	Iteraciones	
A-1	I-1	P-1
A-2	I-2	P-2
A-3	I-3	P-3
A-4	I-4	P-4
A-5	I-5	P-5
A-6	I-6	P-6
A-7	I-7	P-7
A-8	I-8	P-8
A-9	I-9	P-9
A-10	I-10	P-10

➤ **Resultados de Prueba "B"**

- Análisis de precisión

Resultados de Prueba "B"													
N°	L/P	N° de Prueba										N° de Aciertos	Valor en %
		P-1	P-2	P-3	P-4	P-5	P-6	P-7	P-8	P-9	P-10		
1	"A"	√	√	X	√	X	√	X	√	√	√	7	70%
2	"B"	√	X	√	√	√	√	√	√	√	X	8	80%
3	"C"	√	√	√	√	√	√	√	√	√	√	10	100%
4	"D"	X	√	√	√	√	X	√	√	√	√	8	80%
5	"E"	√	X	√	X	√	√	√	X	X	X	5	50%
6	"F"	√	X	X	X	√	√	X	√	√	√	6	60%
7	"G"	√	√	X	√	√	X	√	√	X	X	6	60%
8	"H"	X	√	X	√	√	√	X	X	√	X	5	50%
9	"I"	√	X	√	√	√	√	X	√	√	√	8	80%
10	"K"	X	√	X	√	X	√	√	√	X	√	6	60%
11	"L"	√	X	√	√	√	√	√	√	√	√	9	90%
12	"M"	X	√	√	X	X	X	√	X	√	X	4	40%
13	"N"	X	X	√	X	√	√	X	X	X	X	3	30%
14	"O"	√	X	√	√	√	√	√	√	√	√	9	90%
15	"P"	X	√	√	√	X	√	√	√	√	X	7	70%
16	"Q"	√	X	√	X	√	√	√	√	X	√	7	70%
17	"R"	√	X	√	X	√	√	√	√	X	√	7	70%
18	"S"	√	√	X	√	X	X	X	√	√	X	5	50%
19	"T"	√	X	√	√	√	X	√	√	√	X	7	70%
20	"U"	√	X	√	√	√	√	√	√	√	√	9	90%
21	"V"	√	X	X	√	√	√	√	√	√	X	7	70%
22	"W"	X	√	√	√	√	√	X	√	X	√	7	70%
23	"X"	√	√	√	√	√	√	√	√	√	√	9	90%
24	"Y"	√	X	√	√	√	X	√	X	√	√	8	80%
Total de Pruebas:		240						Total de aciertos:				167	70%

▪ Análisis de rapidez

Resultados de Prueba "B"												
N°	L/P	N° de Prueba										Tiempo promedio
	Letra	P-1	P-2	P-3	P-4	P-5	P-6	P-7	P-8	P-9	P-10	
1	"A"	3	3	6	4	7	3	6	6	4	7	4.9
2	"B"	4	3	4	3	7	3	4	7	7	5	4.7
3	"C"	4	6	3	3	4	5	6	5	5	3	4.4
4	"D"	3	4	6	3	5	5	4	7	7	5	4.9
5	"E"	3	3	7	4	5	4	3	3	6	5	4.3
6	"F"	4	7	7	3	7	7	7	6	7	6	6.1
7	"G"	7	5	5	7	6	7	7	3	4	6	5.7
8	"H"	4	6	3	7	4	3	4	4	7	3	4.5
9	"I"	6	5	7	5	7	4	6	6	4	7	5.7
10	"K"	5	4	7	4	3	7	3	5	5	3	4.6
11	"L"	3	4	6	7	5	5	7	6	4	3	5
12	"M"	6	7	6	6	8	8	8	8	7	6	7
13	"N"	6	5	7	6	8	5	7	5	8	6	6.3
14	"O"	8	4	3	7	4	6	3	8	5	5	5.3
15	"P"	4	7	7	6	6	5	5	4	7	3	5.4
16	"Q"	4	6	6	3	6	5	4	7	6	6	5.3
17	"R"	7	4	3	4	7	3	4	6	6	7	5.1
18	"S"	8	3	4	7	6	7	4	4	6	3	5.2
19	"T"	7	6	5	7	6	7	4	3	5	3	5.3
20	"U"	4	6	6	4	3	6	3	3	5	6	4.6
21	"V"	5	5	7	7	6	4	7	4	5	5	5.5
22	"W"	5	7	5	5	6	5	7	3	3	3	4.9
23	"X"	4	7	6	6	4	4	6	3	4	4	4.8
24	"Y"	5	6	6	5	4	5	7	6	6	7	5.7
Total de Pruebas:		240			Total de tiempos:							125.2

➤ **Encuesta**



**UNS**  
UNIVERSIDAD  
NACIONAL DEL SANTA

**UNIVERSIDAD NACIONAL DEL SANTA**

**“VISIÓN COMPUTACIONAL PARA LA TRADUCCIÓN EN TIEMPO REAL DEL LENGUAJE DE SEÑAS A TEXTO EN IDIOMA ESPAÑOL”**

**1. Facilidad y claridad del funcionamiento del modelo de Visión Computacional**

- Excelente
- Bueno
- Aceptable
- Regular
- Malo

**2. Aporte al aprendizaje del lenguaje de señas peruano a través del uso del modelo**

- Excelente
- Bueno
- Aceptable
- Regular
- Malo

**3. Satisfacción obtenida al emplear la Visión Computacional como medio de comunicación**

- Excelente
- Bueno
- Aceptable
- Regular
- Malo

**Encuesta desarrollada por:**

- Albino Huertas Eder Alberto
- López Olivos Leticia Teresa

➤ **Resultados de encuestas**

Calificación	Indicadores							
	Facilidad y claridad del funcionamiento del modelo de Visión Computacional		Aporte al aprendizaje del lenguaje de señas peruano a través del uso del modelo		Satisfacción obtenida al emplear la Visión Computacional como medio de comunicación		Total	
	Valor	%	Valor	%	Valor	%	Valor	%
Excelente	7	35%	4	20%	10	50%	21	35%
Bueno	4	20%	11	55%	8	40%	23	38%
Aceptable	8	40%	3	15%	2	10%	13	22%
Regular	1	5%	2	10%	0	0%	3	5%
Malo	0	0%	0	0%	0	0%	0	0%
Total	20	100%	20	100%	20	100%	60	100%
Varianza por Item	0,9763		0,7658		0,4632		Suma de Varianzas	
							2,2053	
Varianza de la suma de los Items					5,1473684			

**Alfa de Cronbach:** 
$$\alpha = \left[ \frac{k}{k-1} \right] \left[ 1 - \frac{\sum_{i=1}^k S_i^2}{S_t^2} \right]$$

Donde:

- $S_i^2$  es la varianza del ítem i,
- $S_t^2$  es la varianza de los valores totales observados
- $k$  es el número de preguntas o ítems.

$$\alpha = \left[ \frac{3}{2} \right] \left[ 1 - \frac{2.2053}{5.1473684} \right] = 0.86$$

Criterios de evaluación del Coeficiente de Cronbach:

- Mayor a 0,9: Excelente
  - Mayor a 0,8 y menor a 0,9 es Bueno
  - Mayor a 0,7 y menor a 0,8 Aceptable
  - Mayor a 0,6 y menor a 0,7 Cuestionable
  - Mayor a 0,5 y menor a 0,6 Pobre
  - Menor a 0,5 es Inaceptable
- 0.86 => Bueno**

➤ Entrevistas





