



**SISTEMA DE VISIÓN INTELIGENTE PARA  
DETECCIÓN DE ETIQUETAS EN CONDUCTORES  
ELÉCTRICOS RECUBIERTOS**

**TESIS**

PARA OBTENER EL GRADO DE

**MAESTRO EN  
MANUFACTURA AVANZADA**

PRESENTA

**ING. JUAN JOSÉ NAVARRO MARTÍNEZ  
ASESOR: DRA. CAROLINA RETA CASTRO**

SAN LUIS POTOSÍ, SLP, NOVIEMBRE DE 2020

# CARTA DE LIBERACIÓN DEL ASESOR



Pachuca, Hidalgo, 19 de agosto de 2020.

Mtro. Geovany González Carlos  
Coordinador Académico de Posgrado  
CIATEQ, A.C.

La abajo firmante, miembro del Comité Tutorial del Ing. Juan José Navarro Martínez, una vez revisado su Proyecto Terminal de tesis, titulado "SISTEMA DE VISIÓN INTELIGENTE PARA DETECCIÓN DE ETIQUETAS EN CONDUCTORES ELÉCTRICOS RECUBIERTOS" **autorizo** que el citado trabajo sea presentado por el alumno para su revisión, con el fin de alcanzar el grado de MAESTRO EN MANUFACTURA AVANZADA.

Sin otro particular por el momento, agradezco la atención prestada.

---

Dra. Carolina Reta Castro  
Asesora Académica  
Investigadora en CONACYT-CIATEQ.  
Cédula: 09122140.  
Doctora en ciencias en el área de  
Ciencias Computacionales.

"2020. Año de Leona Vicario, Benemérita Madre de la Patria"

# CARTA DE LIBERACIÓN DEL REVISOR



Querétaro, Qro. 06 de Noviembre del 2020.

Dra. María Guadalupe Navarro Rojero  
Directora de Posgrado  
CIATEQ, A.C.

Por medio de la presente me dirijo a usted en calidad de Revisor del proyecto terminal del alumno **Juan José Navarro Martínez**, cuyo título es:

**“SISTEMA DE VISIÓN INTELIGENTE PARA DETECCIÓN DE ETIQUETAS EN CONDUCTORES ELÉCTRICOS RECUBIERTOS”**

Después de haberlo leído, corregido e intercambiado información con el alumno, y realizado los cambios que le fueron sugeridos, puede ser autorizada su impresión, a fin de que se inicien los trámites correspondientes para su defensa.

Sin otro particular por el momento, y en espera de que mis sugerencias sean tomadas en cuenta en beneficio del estudiante y la Institución, agradezco la atención prestada.

Atentamente,



José Guadalupe Rico Espino

M. en C. José Guadalupe Rico Espino

“2020, Año Leona Vicario, Benemérita Madre de la Patria”

Av. Masanuales N°23-A, Parque Industrial Bernardo Quintana  
CP: 76248, El Marqués, Qro. [www.ciateq.mx](http://www.ciateq.mx)

## DEDICATORIA

A Leticia, Juan José y Yareli, por su apoyo, comprensión, consejos y amor.

A todos mis familiares y amigos, por sus buenos deseos y consejos.

A Jaroslaw por otorgarme la oportunidad y el apoyo.

A Carolina, por su profesionalismo y entrega.

## **AGRADECIMIENTOS**

A todo el personal docente del posgrado en Manufactura Avanzada de CIATEQ por permitirme obtener nuevos conocimientos, en especial a la Dra. Carolina Reta Castro por toda la asesoría brindada a lo largo de este proyecto; al Consejo Nacional de Ciencia y Tecnología por el apoyo No. 498636 otorgado para la realización de mis estudios de Posgrado; a la compañía Continuous Improvement Management Associates S.C. por el apoyo brindado para el desarrollo de este proyecto de investigación, así como la oportunidad de poder crecer profesionalmente a través de un posgrado; y a mis compañeros del posgrado por hacer del curso una experiencia de aprendizaje amena y sin igual.

Juan José Navarro Martínez

San Luis Potosí, SLP

Agosto de 2020

## RESUMEN

El etiquetado de los cables eléctricos permite identificar distintos cables o lotes de productos y conocer sus características. En el proceso de etiquetado se pueden presentar distintos tipos de errores o defectos en el texto impreso sobre los conductores eléctricos. En esta tesis se propone un sistema de visión inteligente aplicado a las líneas de extrusión en la manufactura de cables, permitiendo realizar la inspección visual del etiquetado del cable de forma automática, a través de tres componentes principales: captura de imágenes; algoritmos de tratamiento de imágenes y OCR; y visualización de los resultados. El software del sistema de inspección desarrollado está basado en algoritmos de procesamiento y análisis de imágenes digitales y fue realizado en cinco etapas. En la primera etapa se realiza el pre procesamiento de las imágenes mediante cuatro métodos propuestos: TBH, basado en operaciones morfológicas; KMN, basado en clusterización; MrRecon, basado en reconstrucción morfológica; y un método basado en la fusión de la validación de los métodos anteriores. En la segunda etapa se realiza la detección de ruido y defectos en el texto mediante un clasificador propuesto denominado S.A.F.E. En la tercera etapa se realiza el reconocimiento óptico de caracteres utilizando el motor Tesseract OCR. En la cuarta etapa se realiza la validación del texto a través de un algoritmo propuesto denominado VALET. En la quinta etapa se realiza la visualización de los resultados obtenidos de las etapas anteriores. La validación del texto utilizando VALET permitió obtener una tasa de error promedio del 26.53% para el método TBH, 19.45% para el método KMN, 14.75% para el método MrRecon y 8.69% para la fusión de los tres métodos. El software desarrollado es robusto y permite detectar distintos defectos, manchas pequeñas o desperfectos en el etiquetado. Mediante el sistema de visualización desarrollado, el software alerta al operador sobre los desperfectos o defectos encontrados, brindándole información para que pueda verificar el resultado de la inspección.

**Palabras clave:** Ingeniería y tecnología; Ingeniería de los ordenadores; Sistemas de reconocimiento de caracteres; Detección de texto en cables, Sistemas de visión.

## **ABSTRACT**

Cable labeling allows to identify different cables or batch of products and to know their characteristics. In the labeling process, different types of errors or defects can occur in the printed text on the electrical conductor. In this thesis, an intelligent vision system applied to extrusion lines in cable manufacturing is proposed. The system allows the visual inspection of cable labeling to be carried out automatically through three main components: image capture, image processing and OCR algorithms, and results visualization. The inspection system software is based on digital image processing and analysis algorithms and was carried out in five stages. In the first stage, pre-processing is carried out and four methods were developed: TBH (based on morphological operations), KMN (based on clustering), MrRecon (based on morphological reconstruction), and a method based on the fusion of the validation of all of them. In the second stage, the text's detection of noise and defects is performed by a proposed classifier called S.A.F.E. In the third stage, the optical character recognition is carried out using the Tesseract OCR engine. In the fourth stage, the text is validated using a proposed algorithm called VALET. In the fifth stage, the results obtained from the previous stages are displayed. The developed software is robust and allows detecting different defects, small ink spots, or flaws in the labeling. The text's validation using VALET allowed to obtain an average error rate of 26.53% for the TBH method, 19.45% for the KMN method, 14.75% for the MrRecon method, and 8.69% for the fusion of these three methods. Through the visualization system, the software alerts the operator about the flaws or errors, providing the operator with information to verify the result of the inspection.

**Keywords:** Engineering and technology, Computer engineering, Character recognition systems, Text detection on cables, Vision systems.

# ÍNDICE DE CONTENIDO

CARTA DE LIBERACIÓN DEL ASESOR .....	i
CARTA DE LIBERACIÓN DEL REVISOR .....	ii
RESUMEN .....	v
ABSTRACT .....	vi
ÍNDICE DE CONTENIDO.....	vii
ÍNDICE DE FIGURAS .....	x
ÍNDICE DE TABLAS .....	xiii
GLOSARIO .....	xiv
CAPÍTULO 1. INTRODUCCIÓN .....	1
1.1 ANTECEDENTES .....	1
1.2 PLANTEAMIENTO DEL PROBLEMA .....	2
1.3 JUSTIFICACIÓN .....	3
1.4 OBJETIVOS.....	4
1.5 HIPÓTESIS .....	5
1.6 ORGANIZACIÓN DEL DOCUMENTO .....	5
CAPÍTULO 2. MARCO TEÓRICO .....	7
2.1 PROCESO DE ETIQUETADO EN LOS CABLES .....	7
2.1.1 Recubrimiento del cable eléctrico .....	7
2.1.2 Métodos de etiquetado .....	8
2.1.3 Comparación de métodos de etiquetado.....	10
2.1.4 Nomenclatura de la inscripción del cable.....	11
2.2 SISTEMAS DE VISIÓN POR COMPUTADORA .....	12
2.3 COMPONENTES DEL SISTEMA DE ADQUISICIÓN DE IMÁGENES .....	13
2.3.1 Iluminación .....	14
2.4 ADQUISICIÓN DE IMÁGENES DE OBJETOS EN MOVIMIENTO.....	14
2.5 PROCESAMIENTO Y ANÁLISIS DE IMÁGENES DIGITALES .....	16
2.5.1 Espacios de color .....	17
2.5.2 Filtros espaciales .....	20
2.5.3 Operaciones morfológicas .....	23
2.5.4 Ecualización de histograma .....	25
2.5.5 Segmentación de texto en imágenes .....	25
2.5.6 Umbralización .....	26
2.5.7 Reconstrucción morfológica por dilatación .....	27
2.6 RECONOCIMIENTO DE PATRONES .....	28
2.6.1 Reconocimiento de texto .....	29
2.6.2 Reconocimiento óptico de caracteres (OCR) .....	29

2.6.3 Tesseract-OCR.....	31
2.7 CLUSTERIZACIÓN.....	37
2.7.1 K- medias [46].....	37
2.8 MÉTRICAS DE EVALUACIÓN.....	37
2.8.1 Clasificación binaria .....	37
2.8.2 Clasificación multiclase .....	38
2.8.3 Precisión y exhaustividad .....	38
2.8.4 F-Measure .....	38
CAPÍTULO 3. ESTADO DEL ARTE EN SISTEMAS DE RECONOCIMIENTO DE TEXTO .....	39
3.1 OPTIMIZACIÓN DE EL PRE PROCESADO DE LA IMAGEN CAPTURADA .....	39
3.2 DETECCIÓN DE PLACAS VEHICULARES.....	39
3.3 INSPECCIÓN VISUAL DE PRODUCTOS.....	41
3.4 RECONOCIMIENTO DE TEXTO EN DOCUMENTOS.....	42
3.5 SISTEMA CAPTCHA .....	45
3.6 COMBINACIÓN DE LAS DIFERENTES TÉCNICAS.....	45
3.7 DISCUSIÓN .....	47
CAPÍTULO 4. MÉTODO PROPUESTO .....	50
4.1 GENERALIDADES DEL SISTEMA DE VISIÓN PROPUESTO .....	50
4.2 SISTEMA PROTOTIPO PARA LA ADQUISICIÓN DE IMÁGENES.....	51
4.3 ALGORITMO DE PRE PROCESAMIENTO .....	53
4.3.1 Método basado en operaciones morfológicas, TBH Engine .....	53
4.3.2 Método basado en clusterizacion, KMN Engine.....	60
4.3.3 Método basado en reconstrucción morfológica, MrRecon Engine .....	64
4.3.4 Fusión de métodos .....	69
4.4 ALGORITMO PARA DETECCIÓN DE RUIDO Y DEFECTOS EN EL TEXTO (S.A.F.E.) .....	69
4.5 ALGORITMO DE OCR .....	72
4.6 VALIDACIÓN DEL TEXTO .....	73
4.6.1 Distancia de edición de Levenshtein.....	73
4.6.2 Distancia de edición de Hamming.....	74
4.6.3 Algoritmo de validación del texto, VALET.....	74
4.7 VISUALIZACIÓN DE RESULTADOS .....	77
CAPÍTULO 5. ANÁLISIS EXPERIMENTAL .....	80
5.1 BASE DE DATOS DE IMÁGENES.....	80
5.2 GROUND TRUTH .....	82
5.3 PRUEBAS DE PRE PROCESAMIENTO .....	84
5.3.1 Selección del espacio de color para el método propuesto TBH.....	84
5.3.2 Tamaño y forma del elemento estructural en el método propuesto TBH .....	85
5.3.3 Número de clusters para método propuesto KMN .....	91
5.3.4 Elección de reglas en WEKA para el algoritmo propuesto S.A.F.E.....	93
5.4 EVALUACIÓN DEL DESEMPEÑO DEL SISTEMA.....	94

5.5 ANÁLISIS PARA DETERMINAR EL NÚMERO DE SECUENCIAS NECESARIAS PARA LA VALIDACIÓN DE ETIQUETAS POR EL ALGORITMO VALET .....	101
5.6 DISCUSIÓN DE RESULTADOS .....	107
CONCLUSIONES .....	110
APORTACIÓN DE LA TESIS .....	115
APORTACIÓN SOCIAL DE LA TESIS .....	116
RECOMENDACIONES .....	117
REFERENCIAS BIBLIOGRÁFICAS .....	118

## ÍNDICE DE FIGURAS

FIGURA 1. EJEMPLO DE ETIQUETADO POR CIJ [7] .....	8
FIGURA 2. EJEMPLO DEL MARCAJE POR LÁSER [10].....	9
FIGURA 3. EJEMPLO DE MARCAJE POR TRANSFERENCIA TÉRMICA [11] .....	9
FIGURA 4. EJEMPLO DE IMPRESIÓN POR CONTACTO .....	10
FIGURA 5. SIGNIFICADO DE LA NOMENCLATURA DE LA INSCRIPCIÓN DEL CABLE [13] .....	12
FIGURA 6. COMPONENTES DE UN SISTEMA DE VISIÓN FUENTE: ELABORACIÓN PROPIA.....	13
FIGURA 7. DEFORMACIÓN POR ROLLING SHUTTER. FUENTE: ELABORACIÓN PROPIA .....	15
FIGURA 8. COMPARACIÓN DE IMAGEN EN ESCALA DE GRIS CON RGB [29] .....	17
FIGURA 9. ESPACIO DE COLOR DE LA ESCALA DE GRIS [29] .....	18
FIGURA 10. ESPACIO DE COLOR RGB [29] .....	18
FIGURA 11. REPRESENTACIÓN DE TONO PARA HSV [29] .....	19
FIGURA 12. REPRESENTACIÓN DEL ESPACIO CIE L*A*B* [30].....	20
FIGURA 13. EJEMPLO DE UN KERNEL DE 3 X 3 [31] .....	21
FIGURA 14. (A) IMAGEN CON RUIDO; (B) IMAGEN CON UN FILTRADO DE MEDIANA [32].....	22
FIGURA 15. SALIDA DEL FILTRO LINEAL DE SALIDA [32] .....	22
FIGURA 16. (A) IMAGEN ORIGINAL; (B) IMAGEN AFILADA [32].....	23
FIGURA 17. OPERACIONES MORFOLÓGICAS Y SU COMBINACIÓN [33] .....	24
FIGURA 18. (A) IMAGEN ORIGINAL Y SU HISTOGRAMA; (B)IMAGEN E HISTOGRAMA ECUALIZADOS [34] ...	25
FIGURA 19. (A) IMAGEN CON ILUMINACIÓN DESIGUAL; (B) Y (C) UMBRALIZADO GLOBAL; (D) UMBRALIZADO LOCAL [32] .....	27
FIGURA 20. RECONSTRUCCIÓN MORFOLÓGICA POR DILATACIÓN DE TAMAÑO CINCO DE UNA SEÑAL G CON RESPECTO A UNA SEMILLA F [37] .....	28
FIGURA 21. LAS DIFERENTES ÁREAS DEL OCR [39] .....	30
FIGURA 22. EJEMPLO DE UNA LÍNEA DE TEXTO CON BASE CURVA, LA LÍNEA AZUL REPRESENTA LA LÍNEA BASE AJUSTADA, LA LÍNEA VERDE REPRESENTA LA LÍNEA DESCENDIENTE, LA LÍNEA VIOLETA REPRESENTA LA LÍNEA MEDIA, LA LÍNEA CIAN REPRESENTA UNA LÍNEA ASCENDIDA, LA LÍNEA NEGRA CORRESPONDE UNA LÍNEA RECTA QUE PERMITE OBSERVAR LA PEQUEÑA CURVATURA EN LA LÍNEA CIAN [43] .....	32
FIGURA 23. PALABRA CORTADA POR TONO FIJO [43] .....	32
FIGURA 24. PUNTOS DE CORTE Y LÍNEAS DE CORTE [43].....	33
FIGURA 25. PALABRA CON CARACTERES ROTOS [43].....	33
FIGURA 26. ETAPAS DEL MOTOR DE OCR TESSERACT[43].....	36
FIGURA 27. ESQUEMA DEL PROCESAMIENTO Y ANÁLISIS DE IMÁGENES DEL SISTEMA DE VISIÓN PARA LA INSPECCIÓN DE ETIQUETAS DE CONDUCTORES ELÉCTRICOS FUENTE: ELABORACIÓN PROPIA .	50
FIGURA 28. (A) IMAGEN CAPTURADA CON UNA CÁMARA CON OBTURADOR POR BARRIDO; (B) IMAGEN CAPTURADA CON UNA CÁMARA CON OBTURADOR GLOBAL FUENTE: ELABORACIÓN PROPIA .....	52
FIGURA 29. RESULTADO DE LA OPERACIÓN MORFOLÓGICA Y LA IMAGEN ORIGINAL. (A) IMAGEN RGB ANTES DE APLICAR TOP-HAT; (B) IMAGEN RGB ANTES DE APLICAR BLACK HAT; (C) RESULTADO DE TOP-HAT; (D) RESULTADO DE BLACK-HAT FUENTE: ELABORACIÓN PROPIA....	56

FIGURA 30. IMAGEN BINARIZADA POR EL MÉTODO DE OTSU FUENTE: ELABORACIÓN PROPIA .....	57
FIGURA 31. CONJUNCIÓN LÓGICA AND UTILIZANDO UNA MÁSCARA FUENTE: ELABORACIÓN PROPIA ..	58
FIGURA 32. (A) MÁSCARA OBTENIDA POR LA ECUACIÓN 11; (B) MÁSCARA OBTENIDA POR LAS ECUACIONES 12, 13 Y 14 FUENTE: ELABORACIÓN PROPIA.....	59
FIGURA 33 – REDUCCIÓN DEL BORDE EN LA IMAGEN CON TEXTO SEGMENTADO. (A) IMAGEN <i>IMGBW</i> ; (B) BORDE MEJORADO EN IMAGEN ANTERIOR FUENTE: ELABORACIÓN PROPIA .....	60
FIGURA 34. RESULTADOS DEL CLUSTERIZADO, CADA GRUPO ESTÁ REPRESENTADO CON UN COLOR DISTINTO FUENTE: ELABORACIÓN PROPIA.....	61
FIGURA 35. (A) MÁSCARA OBTENIDA POR LA ECUACIÓN 16 CUANDO EL FONDO ES CLARO; (B) MÁSCARA OBTENIDA POR LA ECUACIÓN 17 CUANDO EL FONDO ES OSCURO FUENTE: ELABORACIÓN PROPIA .....	63
FIGURA 36. RESULTADO DEL ALGORITMO DE CLUSTERIZACIÓN FUENTE: ELABORACIÓN PROPIA .....	64
FIGURA 37. (A) IMAGEN ORIGINAL; (B) IMAGEN RESULTANTE DE LA RECONSTRUCCIÓN MORFOLÓGICA POR DILATACIÓN FUENTE: ELABORACIÓN PROPIA.....	67
FIGURA 38. RESULTADO DE LA RESTA DE LAS IMÁGENES EN LA ECUACIÓN 23, LA CUAL PERMITE SEGMENTAR EL TEXTO DE LA IMAGEN ORIGINAL FUENTE: ELABORACIÓN PROPIA .....	68
FIGURA 39. REPRESENTACIÓN DE LAS TRES CLASES DE CONTORNOS EN UNA SOLA IMAGEN FUENTE: ELABORACIÓN PROPIA .....	70
FIGURA 40. REGLAS PARA CLASIFICAR LOS CONTORNOS FUENTE: ELABORACIÓN PROPIA.....	71
FIGURA 41. REPRESENTACIÓN GRÁFICA DEL ALGORITMO DE EXTRACCIÓN DE ERRORES Y RUIDO FUENTE: ELABORACIÓN PROPIA .....	72
FIGURA 42. SECUENCIA DE IMÁGENES Y SU VALIDACIÓN A TRAVÉS DEL ALGORITMO VALET. (A) PRIMERA IMAGEN; (B) SEGUNDA IMAGEN; (C) TERCERA IMAGEN; (D) CUARTA IMAGEN FUENTE: ELABORACIÓN PROPIA .....	76
FIGURA 43. RESULTADO DEL MOTOR OCR FUENTE: ELABORACIÓN PROPIA .....	77
FIGURA 44. CÓDIGO DE COLORES. (A) TEXTO CORRECTO CONSIDERANDO SOLAMENTE LOS CARACTERES; (B) TEXTO CORRECTO UTILIZANDO UN COMODÍN; (C) TEXTO INCORRECTO; (D) TEXTO CORRECTO. FUENTE: ELABORACIÓN PROPIA .....	79
FIGURA 45. MUESTRA DE IMÁGENES QUE CONFORMAN LA BASE DE DATOS. (A) VIAKON NEGRO; (B) VIAKON BLANCO; (C) KOBREX ROJO; (D) VIAKON UL; (E) INDIANA ROJO; (F) VIAKON VERDE; (G) KOBREX AZUL; (H) KOBREX VINIKOB FUENTE: ELABORACIÓN PROPIA.....	82
FIGURA 46. IMAGEN CON COMODÍN FUENTE: ELABORACIÓN PROPIA .....	83
FIGURA 47. COMPARACIÓN DE DISTINTOS ELEMENTOS ESTRUCTURALES. (A) RECT, CROSS Y ELLIPSE 11x11; (B) RECT, CROSS Y ELLIPSE 17x17; (C) RECT, CROSS Y ELLIPSE 21x21; (D) RECT, CROSS Y ELLIPSE 31x31; (E) RECT, CROSS Y ELLIPSE 51x51; (F) RECT, CROSS Y ELLIPSE 81x81 FUENTE: ELABORACIÓN PROPIA .....	86
FIGURA 48. COMPARACIÓN DE DISTINTOS ELEMENTOS ESTRUCTURALES. (A) RECT, CROSS Y ELLIPSE 11x11; (B) RECT, CROSS Y ELLIPSE 17x17; (C) RECT, CROSS Y ELLIPSE 21x21; (D) RECT, CROSS Y ELLIPSE 31x31; (E) RECT, CROSS Y ELLIPSE 51x51; (F) RECT, CROSS Y ELLIPSE 81x81 FUENTE: ELABORACIÓN PROPIA .....	88

FIGURA 49. COMPARACIÓN DEL ELEMENTO ESTRUCTURAL PROPUESTO. (A) E. S. PROPUESTO 11x11; (B) E. S. PROPUESTO 17x17; (C) E. S. PROPUESTO 21x21; (D) E. S. PROPUESTO 31x31; (E) E. S. PROPUESTO 51x5; (F) E. S. PROPUESTO 81x81 FUENTE: ELABORACIÓN PROPIA .....	90
FIGURA 50. ERRORES AL UTILIZAR 3 CLUSTERS EN EL MÉTODO KMN DENTRO DE DISTINTAS COLECCIONES. (A) VIAKON UL; (B) KOBREX AZUL; (C) KOBREX VINIKOB; (D) VIAKON UL; (E) KOBREX ROJO FUENTE: ELABORACIÓN PROPIA.....	92
FIGURA 51. RESULTADO AL UTILIZAR 4 CLUSTERS CON EL MÉTODO KMN DENTRO DE DISTINTAS COLECCIONES. (A) VIAKON UL; (B) KOBREX AZUL; (C) KOBREX VINIKOB; (D) VIAKON UL; (E) KOBREX ROJO FUENTE: ELABORACIÓN PROPIA .....	93
FIGURA 52. ACIERTOS POR COLECCIÓN DE IMÁGENES, EVALUANDO CADA UNO DE LOS MÉTODOS PROPUESTOS Y SU COMBINACIÓN FUENTE: ELABORACIÓN PROPIA .....	97
FIGURA 53. VALIDACIÓN DE ETIQUETAS POR NÚMERO DE SECUENCIAS, COMPARANDO EL NÚMERO DE SECUENCIAS CONTRA EL PORCENTAJE DE ERROR. (A) CORRESPONDE AL GRUPO DE IMÁGENES DEL CABLE INDIANA ROJO; (B) CORRESPONDE AL GRUPO DE IMÁGENES DEL CABLE VIAKON BLANCO; (C) CORRESPONDE AL GRUPO DE IMÁGENES DEL CABLE KOBREX AZUL; (D) CORRESPONDE AL GRUPO DE IMÁGENES DEL CABLE KOBREX ROJO; (E) CORRESPONDE AL GRUPO DE IMÁGENES DEL CABLE KOBREX VINIKOB; (F) CORRESPONDE AL GRUPO DE IMÁGENES DEL CABLE VIAKON NEGRO; (G) CORRESPONDE AL GRUPO DE IMÁGENES DEL CABLE VIAKON UL; Y (H) CORRESPONDE AL GRUPO DE IMÁGENES DEL CABLE VIAKON VERDE FUENTE: ELABORACIÓN PROPIA.....	103
FIGURA 54 – NIVEL DE EFICACIA MÉTODO VERSUS PRODUCTO FUENTE: ELABORACIÓN PROPIA.....	108

## ÍNDICE DE TABLAS

TABLA 1. VENTAJAS Y DESVENTAJAS DE LOS MÉTODOS DE IMPRESIÓN PARA REALIZAR RECONOCIMIENTO DE CARACTERES .....	10
TABLA 2. COMPARACIÓN DE OPERACIONES MORFOLÓGICAS APLICADAS EN IMÁGENES BINARIAS Y EN IMÁGENES A ESCALA DE GRISES .....	24
TABLA 3. TRABAJO RELACIONADO VERSUS TRABAJO PROPUESTO .....	49
TABLA 4. DATOS OBTENIDOS DEL MOTOR DE OCR EN EL SISTEMA PROPUESTO .....	73
TABLA 5. BASE DE DATOS PARA DESARROLLO DEL ANÁLISIS EXPERIMENTAL .....	80
TABLA 6. COMPARACIÓN ENTRE ESPACIOS DE COLORES PARA MÉTODO TBH .....	84
TABLA 7. ACIERTOS POR CONJUNTO UTILIZANDO LOS LÍMITES PARA EL ELEMENTO ESTRUCTURAL PROPUESTO .....	91
TABLA 8. COMPARACIÓN ENTRE CLASIFICADORES BASADOS EN REGLAS .....	93
TABLA 9. RESULTADOS DE LA EVALUACIÓN DEL DESEMPEÑO DEL SISTEMA POR COLECCIÓN Y MÉTODO, CORRESPONDIENDO CADA RESULTADO AL PORCENTAJE DE ACIERTOS OBTENIDOS. ....	95
TABLA 10. PORCENTAJE DE ERROR EN LAS DISTINTAS COLECCIONES .....	100

## GLOSARIO

Símbolo	Descripción
OCR	Reconocimiento óptico de caracteres (Optical Character Recognition)
GT	Base verdadera (Ground Truth)
KMN	K-medias (K-Means)
TBH	Transformación sombrero de copa blanco o copa negra (Top Black Hat)
VALET	VALidación de Etiquetas
S.A.F.E.	Detector de ruido y fallas (noise and fault detector)
E. S.	Elemento estructural

# CAPÍTULO 1. INTRODUCCIÓN

## 1.1 ANTECEDENTES

“La última revolución industrial llamada Industria 4.0 (I4.0) representa una evolución en los procesos de fabricación al incorporar las nuevas metodologías digitales de la actualidad. Hoy en día, es indispensable para cualquier compañía industrial sumarse a dicha evolución para no perder su nivel de competitividad y evitar quedar en breve fuera del mercado.” [1] Dentro de las herramientas que permiten implementar estas nuevas metodologías se encuentran los sistemas de visión artificial. Las características que los sistemas de visión artificial otorgan a la I4.0 son: “la trazabilidad, el control de calidad, el soporte a la producción, la seguridad industrial, el control de procesos, la logística, así como la generación de enormes cantidades de datos” [1]. Los sistemas de visión al trabajar las veinticuatro horas del día buscan generar estadísticas y datos precisos del proceso en todo momento.

La visión artificial surgió en la década de los años 60 con la idea básica de conectar una cámara de video a un computador, incluyendo no solo el proceso de captura de las imágenes a través de la cámara sino también abarcando el proceso de procesamiento y análisis de las imágenes, el cual permite interpretar o comprender la información visual que está presente en el mundo [2]. Dos actividades básicas de la visión artificial son: el procesamiento de imágenes digitales y la visión computacional; la primera tiene como objetivo la descripción y reconocimiento del contenido de una imagen digital y la segunda busca dotar a los ordenadores la capacidad de poder simular la visión humana [3]. Todo estudio de visión artificial debe incluir el cómo extraer de las imágenes, los aspectos del mundo que son útiles para tomar decisiones; creando así una necesidad del hombre por construir máquinas que realicen el procesado de la información.

En el sector industrial de la fabricación de conductores eléctricos poco a poco se van implementando herramientas tecnológicas para lograr su conversión en una I4.0. Dentro de los diferentes procesos que se realizan para la fabricación de cable eléctrico existe el proceso de extrusión[4], el cual se entiende como la aplicación de un recubrimiento plástico o goma sobre un conductor eléctrico. Posterior a esto se etiqueta el producto

con base en las especificaciones del fabricante. El proceso de etiquetado puede ser desde grabado por relieve con dados hasta sistemas de impresión por inyección de tinta [5]–[7]. Sin embargo, al depender de tinta o de un dado grabador para el correcto marcaje, en el proceso de etiquetado pueden presentarse problemas como: corrimiento de la tinta, daño en el dado grabador por manipulación incorrecta, impresión errónea del texto debido a una mala limpieza del dado grabador, entre otros.

En esta tesis se estudia el problema de detección y validación automática de etiquetas en conductores eléctricos recubiertos a través de un sistema de visión artificial. Actualmente existen soluciones similares, las cuales llegan a tener un alto costo, baja compatibilidad con distintas soluciones de impresión y se limitan a la detección del texto impreso sin detectar defectos producidos por la impresión. La solución propuesta estará constituida por una unidad de adquisición de imágenes y una unidad de procesamiento y análisis de información; la primera unidad incluirá al sistema de iluminación y captura de imagen. La segunda unidad se encargará de procesar la imagen capturada mediante algoritmos y así transformarla en información de alto nivel. Las características que le otorgan robustez al sistema desarrollado son: la capacidad de capturar el texto deseado considerando la velocidad a la cual el objetivo (cable eléctrico etiquetado) se encuentra; la condición que afecta a la iluminación debido a que se presenta una superficie reflejante; la compatibilidad con diversas líneas de producción; la facilidad de ser configurable con base en la solución que requieran los clientes; la capacidad de reconocimiento en distintos colores de cables y colores de etiquetas; y la posibilidad de detectar errores en la impresión así como en el texto impreso. El análisis de estas características evitará problemas dentro del desarrollo de la solución.

Cabe señalar que en este documento no se darán detalles de la unidad de adquisición de imágenes utilizada en el prototipo, pues se considera material confidencial, debido a que será sometido a patente.

## **1.2 PLANTEAMIENTO DEL PROBLEMA**

Actualmente, el proceso de revisión de etiquetado se realiza de manera manual. Este proceso consiste en utilizar una lámpara estroboscópica y sincronizar la velocidad de encendido y apagado con la velocidad de la máquina; esto crea un efecto visual con

el cual el operador puede ver la etiqueta de manera semi estática para poder validar visualmente si la etiqueta está siendo impresa correctamente. Las velocidades de operación de las líneas de extrusión de cable eléctrico se encuentran en un rango entre los 150 y 400 m/min, resultando imposible para el ojo humano la visualización de la etiqueta sin realizar el proceso manual mencionado.

La problemática principal relacionada con el etiquetado en estas líneas de producción se presenta cuando hay ausencia de etiquetas, implicando la necesidad de volver a realizar el etiquetado del cableado y, por consecuencia, aumentando el tiempo de producción en la línea. Por otro lado, otra de las problemáticas es la presencia de la etiqueta con información errónea, generando el tener que desechar el producto por completo o tener que pasarlo a un reprocesado, dependiendo del producto que se esté fabricando. Estas problemáticas aumentan el tiempo de fabricación del producto, generando una alteración dentro de la planificación de producción. Esto conlleva el tener que realizar acciones fuera de las consideradas para la fabricación del producto, atrasando a su vez los tiempos de producción de otras líneas o generando desperdicio de material en casos donde no es posible realizar un reprocesado.

Poder realizar la detección y la validación de manera automática de las etiquetas que se graban en el conductor eléctrico, el cual está siendo fabricado por una máquina extrusora, resuelve el principal problema de la ausencia de esta característica, permitiendo a su vez cumplir con estándares de calidad para el producto.

En esta tesis se abordarán las distintas dificultades que se pueden presentar en el proceso de captura de las imágenes: Los efectos de la iluminación en las imágenes, la velocidad del objeto a capturar, reflejos en el objeto, variaciones en la impresión.

### **1.3 JUSTIFICACIÓN**

El desarrollar un sistema de visión inteligente capaz de realizar la detección de etiquetas en las líneas extrusoras de cable eléctrico repercute en una mejora en el control de producción y calidad para este proceso de fabricación; debido a que detectando de una manera rápida y precisa errores en el etiquetado se pueden tomar acciones

correctivas en el producto, evitando así que este tenga que ser desechado por completo o reprocesado.

Esto le dará al sistema de control de la línea la habilidad de controlar que el producto cumpla con estándares de calidad y que los niveles de producción no se verán afectados por problemas con el etiquetado, generando así un impacto positivo en la economía del fabricante.

La integración de diferentes tecnologías en un solo sistema repercute en un avance tecnológico, ya que los sistemas de visión actuales cuentan con ciertas limitantes, tendiendo a requerir condiciones de operación muy controladas para su correcto funcionamiento. Lo que se expone en esta tesis, dicta un nuevo enfoque a los sistemas de visión en la industria de los conductores. En esta industria, las aplicaciones con sistemas de visión varían por el proceso de fabricación o por el fabricante de las máquinas, inclusive en algunos casos pueden llegar a ser nulas.

## **1.4 OBJETIVOS**

### **Objetivo general**

Diseñar y desarrollar un sistema de visión que integre un módulo de adquisición de imágenes y algoritmos para la detección de etiquetas en conductores eléctricos recubiertos, que sea robusto a deformaciones y cambios en el color del cable, así como a variaciones en las características del rótulo.

### **Objetivos específicos**

1. Integrar un módulo de adquisición de imágenes a un computador, con iluminación uniforme, que utilice protocolos estándar de comunicación y con parámetros de captura configurables.
2. Diseñar un algoritmo de segmentación que permita identificar y señalar las diferencias encontradas en el cable, que sea robusto a cambios de color y producto con base en la similitud que se presenta en la etiqueta.

3. Diseñar un algoritmo para la detección de rótulos y validación del texto, a partir de la aplicación de algoritmos para el reconocimiento óptico de caracteres.
4. Integrar métricas para la validación del texto obtenido en el reconocimiento óptico de caracteres mediante su comparación contra el texto real utilizando métricas de distancia de texto.

En este documento no se darán detalles del módulo de adquisición de imágenes utilizada en el prototipo, pues se considera material confidencial, debido a que será sometido a patente.

## **1.5 HIPÓTESIS**

Por medio de técnicas basadas en visión artificial se demostrará la factibilidad de poder segmentar caracteres, extraer defectos o ruido en la impresión de texto y validar las etiquetas en el cable. La validación automática del texto impreso en conductores eléctricos recubiertos permitirá detectar errores de forma temprana como gotas de tinta y letras con segmentos faltantes los cuales se pueden presentar durante el proceso de etiquetado.

## **1.6 ORGANIZACIÓN DEL DOCUMENTO**

La tesis está organizada de la siguiente manera.

En el capítulo 2 se describe el marco teórico, donde se explican los conceptos fundamentales sobre: sistemas de etiquetado en conductores eléctricos, sistemas de visión, reconocimiento de patrones y métricas de evaluación

En el capítulo 3 se presenta una revisión del estado del arte en sistemas de reconocimiento de texto presentados en artículos de divulgación científica.

En el capítulo 4 se expone el método propuesto para la solución del problema, detallando las etapas del algoritmo desarrollado.

En el capítulo 5 se presentan los detalles del análisis experimental realizado para validar y evaluar el método propuesto, una descripción de la base de datos utilizada, la

generación y descripción de la ground truth (en español, base verdadera), así como la manera en la que se seleccionaron ciertos parámetros del algoritmo de la etapa de pre procesamiento. También se presenta la evaluación del desempeño del sistema de visión sobre la base de datos.

En el capítulo 6 se presentan las conclusiones del trabajo desarrollado, así como el trabajo futuro propuesto.

Por último, se enlistan las referencias utilizadas en esta tesis.

## **CAPÍTULO 2. MARCO TEÓRICO**

En este capítulo se presenta teoría relevante para fundamentar el trabajo de tesis, incluyendo una explicación del proceso de etiquetado en los cables, así como conceptos clave sobre los sistemas de visión por computadora, el procesamiento y análisis de imágenes digitales, el reconocimiento de patrones y métricas de evaluación, todos ellos aplicados a la detección de etiquetas.

### **2.1 PROCESO DE ETIQUETADO EN LOS CABLES**

En los distintos procesos de fabricación de los conductores eléctricos existe un proceso que consiste en colocar una etiqueta o leyenda sobre el recubrimiento del conductor; el método que se utiliza para realizar la colocación de la etiqueta puede variar dependiendo del tipo de recubrimiento o del proceso de etiquetado que se necesita, generando en los distintos procesos diferentes características en el texto impreso.

#### **2.1.1 Recubrimiento del cable eléctrico**

Los recubrimientos que son aplicados a un cable eléctrico proporcionan protección eléctrica o mecánica; dentro de estos se encuentra el de aislamiento, en el cual se coloca un recubrimiento aislante sobre el conductor para evitar fugas de corriente, clasificándolos en dos grandes grupos: termoplásticos y termoestables [8].

“Los recubrimientos termoplásticos más usuales son: PVC (Policloruro de vinilo), PE (Polietileno lineal), Teflón (fluorados)” [8] , Nylon, Poliuretano (PU) [9]. “A su vez, los aislamientos termoestables más usuales son: EPR (Etileno Propileno), XLPE (Polietileno Reticulado), SI (Silicona), PCP (Neopreno)” [8], SBR (caucho de Estireno-butadieno), NR (Caucho natural), IIR (caucho butílico), EVA (Acetato de Etil Vinil), CR o PCP (caucho de cloropreno), CSP o CSM (caucho de polietileno clorosulfonado)[9].

Para las pruebas del método que se propone en este trabajo se utilizaron cables con recubrimiento de PVC.

## 2.1.2 Métodos de etiquetado

Dentro de las tecnologías que permiten el etiquetado o marcado de los cables existen diferentes alternativas, las cuales se presentan a continuación.

### **Sistema de chorro de tinta continua (CIJ, siglas en inglés de continuous inkjet)**

Método sin contacto que crea un chorro de pequeñas gotas de tinta y selecciona algunas para realizar la impresión, convirtiendo estas gotas en caracteres, esparciendo la tinta a través del aire gota a gota. La Figura 1 presenta un ejemplo de texto impreso por CIJ.



Figura 1. Ejemplo de etiquetado por CIJ [7]

### **Marcaje láser**

Método en el cual se utiliza comúnmente un láser de CO<sub>2</sub> para realizar el grabado de la etiqueta sobre el conductor. La Figura 2 presenta unos ejemplos de texto impreso por marcaje láser en distintos cables.



Figura 2. Ejemplo del marcaje por láser [10]

### **Transferencia térmica**

Método en el cual se genera una imagen en una cinta de color y esta se transfiere posteriormente al objeto a través de un rodillo de transferencia. La Figura 3 presenta un ejemplo de texto impreso por transferencia térmica en un par de cables.



Figura 3. Ejemplo de marcaje por transferencia térmica [11]

### **Impresión por contacto**

Método en el cual los caracteres son pregrabados en una rueda de impresión de precisión y se pone en contacto con la superficie del material generando la impresión

del texto, el cual puede contener tinta para mejorar la impresión. La Figura 4 presenta un ejemplo de marcaje por contacto en un cable.



Figura 4. Ejemplo de impresión por contacto

Fuente: elaboración propia

### 2.1.3 Comparación de métodos de etiquetado

En la Tabla 1 se presentan las ventajas y desventajas de los métodos de impresión para el reconocimiento de caracteres.

Tabla 1. Ventajas y desventajas de los métodos de impresión para realizar reconocimiento de caracteres

<b>Método</b>	<b>Ventaja</b>	<b>Desventaja</b>
CIJ	Alta repetibilidad en la impresión.	Espaciado en los caracteres.
Láser	Claridad en la impresión.	Posibilidad de quemar el texto en el material, resultando complicado su lectura.
Transferencia térmica	Fácil mantenimiento, lo cual garantiza la repetibilidad en la impresión.	Depende de una cinta térmica, la cual puede dejar de imprimir correctamente el texto.
Contacto	Buena repetibilidad en la impresión y fácil mantenimiento.	Necesita estar en contacto con el material, lo cual en algunos productos no es muy recomendado. Si no se tiene el mantenimiento adecuado existe el riesgo de que la impresión de la leyenda no sea continua.

En este trabajo se estudia específicamente la detección de etiquetas en conductores eléctricos recubiertos con marcaje por contacto, ya que este método se utiliza con

frecuencia en las líneas de extrusión de cable, debido a que requiere poco mantenimiento; al ser un método en el cual el contacto del dado grabador con el cable es el que realiza el marcaje de la etiqueta nos permite tener la certeza de que la impresión se está realizando en el producto. El texto impreso con este método presenta una baja variación durante el proceso de etiquetado, manteniendo el mismo patrón de impresión. Sin embargo, al depender de tinta o de un dado grabador para el correcto marcaje, en el proceso de etiquetado pueden presentarse problemas como: corrimiento de la tinta; daño en el dado grabador por manipulación incorrecta; impresión errónea del texto debido a una mala limpieza del dado grabador; entre otros.

#### **2.1.4 Nomenclatura de la inscripción del cable**

La National Electricity Code (NEC o NFPA 70) proporciona en el artículo 310.4, específicamente en la tabla 310.4(A), un resumen de los tipos de cable dentro del estándar versión 2020 [12]. Con base en este estándar se puede identificar el tipo de aislamiento que tiene un cable en la inscripción o etiqueta que aparece sobre él. La nomenclatura de la inscripción se divide principalmente en [13]:

- 1- **“T (Thermoplastic)**. Aislamiento termoplástico (este lo tienen todos los cables sin importar la composición del recubrimiento)” [13] .
- 2- **“H (Heat resistant)**. Resistente al calor hasta 75° centígrados (167° F)” [13].
- 3- **“HH (Heat resistant)**. Resistente al calor hasta 90° centígrados (194° F)” [13].
- 4- **“W (Water resistant)**. Resistente al agua y a la humedad” [13].
- 5- **“LS (Low smoke)**. Baja emisión de humos y bajo contenido de gases contaminantes” [13].
- 6- **“SPT (Service parallel thermoplastic)**. Se usa para identificar un cordón que se compone de dos cables paralelos con aislamiento de plástico y que están unidos entre sí. También se denomina cordón dúplex” [13].

En la Figura 5 se muestran dos ejemplos de cables etiquetados con la nomenclatura anteriormente descrita y se presentan otros parámetros que son explicados a continuación.

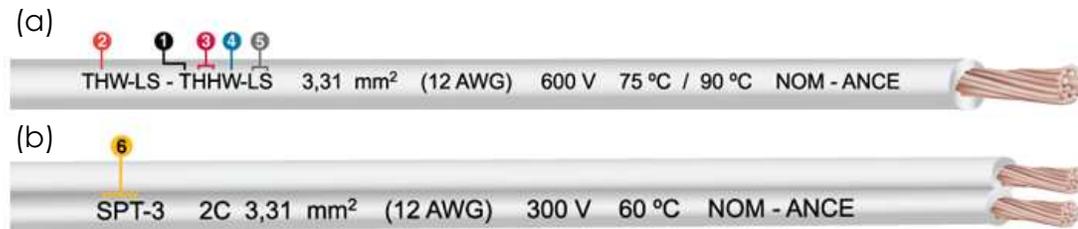


Figura 5. Significado de la nomenclatura de la inscripción del cable [13]

En la Figura 5, dentro de los parámetros que son impresos en el cable también se encuentran el del diámetro del conductor de cobre en mm<sup>2</sup>, seguido por el calibre del cable en base a la AWG (American Wire Gauge) [14] y otros datos como la tensión máxima de trabajo, temperaturas de trabajo y las entidades que norman los cables. El formato final de las etiquetas no está completamente establecido, ya que quien define esto es el cliente final de cada fabricante. La etiqueta de la Figura 5(a) describe a un cable termoplástico que es resistente al calor y al agua, es de baja emisión de humos, tiene un conductor de 3,31 mm<sup>2</sup> de diámetro, calibre 12 en AWG, soporta hasta 600 volts, su temperatura de trabajo es de 75 °C mojado y 90 °C seco y se rige por la normativa NOM-ANCE [15]. La etiqueta de la Figura 5(b) describe a un cable compuesto por dos cables paralelos con aislamiento termoplástico formado por dos conductores de 3.31 mm<sup>2</sup> de diámetro, calibre 12 en AWG, con una tensión máxima de 300 volts, a una temperatura de 60 °C y que se rige por la normativa NOM-ANCE [15].

## 2.2 SISTEMAS DE VISIÓN POR COMPUTADORA

La visión computacional permite estudiar y entender el proceso de la visión humana para poder construir máquinas con capacidades similares [16].

El objetivo universal de la visión computacional es proveer información visual. Para lograr esto, los sistemas de visión pueden ser comprendidos por diferentes tareas agrupadas en tres categorías [17]:

- Visión de bajo nivel (adquisición de imágenes y pre procesamiento).
- Visión de medio nivel (segmentación, representación y clasificación).
- Visión de alto nivel (interpretación y entendimiento conceptual de la información visual).

La Figura 6 presenta en un diagrama los componentes básicos que forman un sistema de visión agrupados en los tres niveles mencionados anteriormente. La relación entre componentes se representa con una flecha indicando el flujo de datos, excepto en el sistema de iluminación el cual busca mejorar las características del objeto para el sistema de adquisición de imágenes.

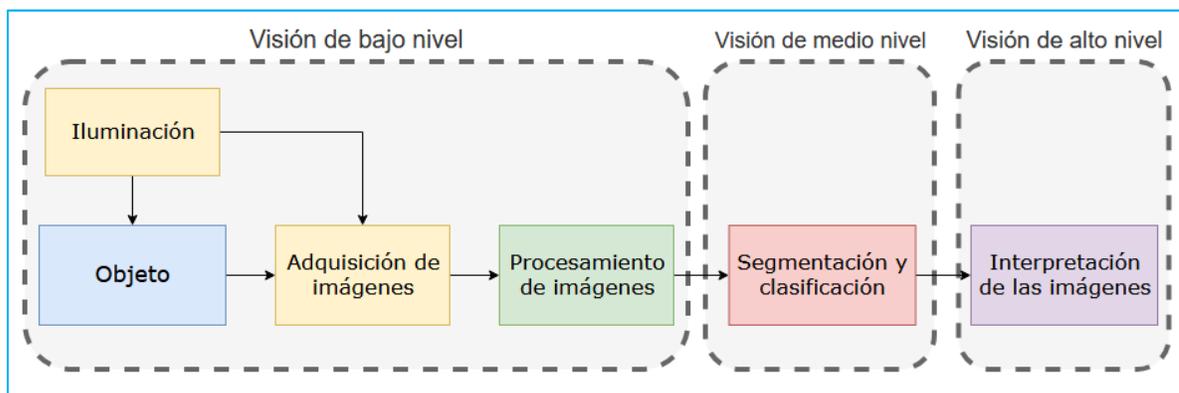


Figura 6. Componentes de un sistema de visión

Fuente: elaboración propia

### 2.3 COMPONENTES DEL SISTEMA DE ADQUISICIÓN DE IMÁGENES

“El proceso de formación y captura de una imagen depende de un conjunto de elementos, cada uno de los cuales cumple una misión determinada. Básicamente, el proceso consiste en generar una señal electromagnética en forma de luz que se transmite, refracta y refleja en la escena hasta que alcanza el área del sensor de la cámara. En el sensor se produce la transducción a una señal eléctrica y a partir de ese momento se dispone de un conjunto de elementos electrónicos que digitalizan, formatean, transmiten y almacenan las imágenes digitales” [18].

Dentro de los componentes que forman el sistema de adquisición de imágenes se encuentran las fuentes de iluminación, la cámara con la que se obtienen las imágenes y los lentes que complementan a la cámara. "Los elementos electrónicos que se encuentran a partir de la cámara son la electrónica de la cámara y los cables de transmisión para un determinado protocolo de comunicación" [18].

### **2.3.1 Iluminación**

La iluminación es un factor importante en la resolución de la aplicación requerida [19], debido a que las cámaras captan la luz reflejada de los objetos para su posterior análisis [19]. El propósito de la iluminación es el de controlar la forma en la cual la cámara finalmente percibe el objeto a analizar. La luz se refleja de diferente manera en cada superficie [20], por lo cual resulta importante ajustar el sistema al objeto a iluminar y su reflectividad; el utilizar la iluminación de manera correcta o de manera inapropiada puede ser un factor decisivo para facilitar o dificultar la aplicación a solucionar.

En este documento no se darán detalles de la iluminación utilizada en el prototipo de captura de imágenes digitales, pues se considera material confidencial, debido a que será sometido a patente.

## **2.4 ADQUISICIÓN DE IMÁGENES DE OBJETOS EN MOVIMIENTO**

En el proceso de formación de imágenes son los transductores los que convierten la radiación luminosa reflejada en señales eléctricas [21]. Cuando una imagen es capturada por una cámara, la luz viaja a través de lentes, incidiendo directamente sobre un sensor que convierte la luz en una señal eléctrica. El sensor consiste de pequeños elementos llamados pixeles los cuales registran la cantidad de luz que incide en ellos; mientras más luz, más electrones son generados, los cuales representan finalmente cierto voltaje que se transforma en un valor numérico, todo esto realizándose dentro del circuito electrónico contenido en la cámara. Dentro de las variantes que existen en los sensores están los CMOS (en inglés Complementary Metal Oxide Semiconductor) y los CCD (en inglés Charge-Coupled Device) [22].

En la captura de imágenes existe la posibilidad de que las imágenes no otorguen información suficiente para poder realizar el análisis y reconocimiento de la información que contiene. Por lo tanto, es importante conocer las características de nuestra cámara, así como las de objetivo a capturar. En este proyecto, el sistema de visión detectará el texto que es impreso en un conductor eléctrico recubierto. Por lo tanto, el texto se considerará como el objetivo de análisis, el cual se encuentra en movimiento por el proceso de fabricación del conductor eléctrico. Por lo cual, es posible que se presente una deformación óptica debido al movimiento. Esta deformación óptica es conocida como rolling shutter [23].

En la Figura 7 se presenta un ejemplo de la deformación de un texto por rolling shutter; la imagen de la Figura 7(a) se capturó estando el objeto sin movimiento. Sin embargo, al empezar a moverse el objeto, mientras se está realizando la integración de la imagen en el sensor, la imagen que este percibe (ver Figura 7(b)) presenta una deformación en el texto por rolling shutter.

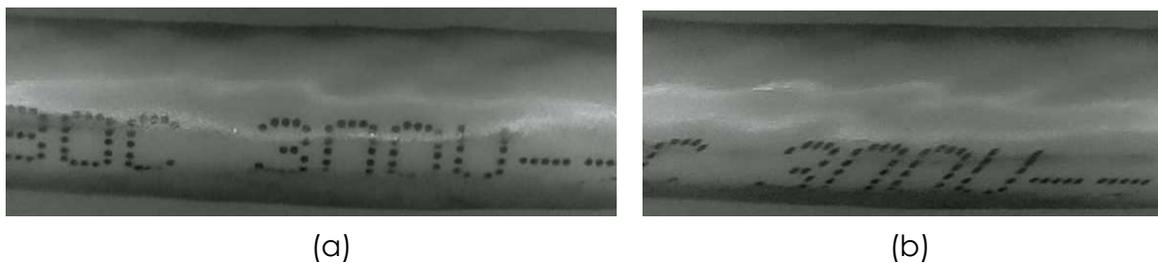


Figura 7. Deformación por rolling shutter.

Fuente: elaboración propia

Para corregir la deformación por rolling shutter se han desarrollado diferentes algoritmos, desde métodos de rectificación y compensación de la imagen [24], [25] hasta métodos en los cuales se verifica cuál de los cuadros obtenidos en una secuencia de captura tiene las mejores características [26].

Una manera de solucionar la distorsión por rolling shutter consiste en cambiar el tipo de obturador de la cámara por uno global shutter [27], [28] permitiendo capturar el objeto en movimiento sin distorsión. Las imágenes utilizadas como base de datos, no presentan deformación por rolling shutter.

En este documento no se darán detalles sobre la selección del tipo de cámara a utilizar en el prototipo de captura de imágenes digitales, pues se considera material confidencial, debido a que será sometido a patente.

## **2.5 PROCESAMIENTO Y ANÁLISIS DE IMÁGENES DIGITALES**

En términos de visión computacional, una imagen es simplemente una matriz, en otras palabras, un vector bidimensional, con un número válido de filas y columnas. Las características de una imagen son:

- El ancho de la imagen, que corresponde al número de columnas en la matriz.
- El alto de la imagen, que corresponde al número de filas en la matriz.
- Cada elemento de la matriz representa un pixel, el cual es el componente más básico de una imagen. Una imagen es una colección de pixeles.
- Cada pixel puede contener uno o más valores que corresponden a una representación visual (por ejemplo: color, brillo, etc.).

En la Figura 8 se muestran dos versiones ampliadas de la misma área de una imagen, estando una en formato de escala de grises y otra en color (RGB).

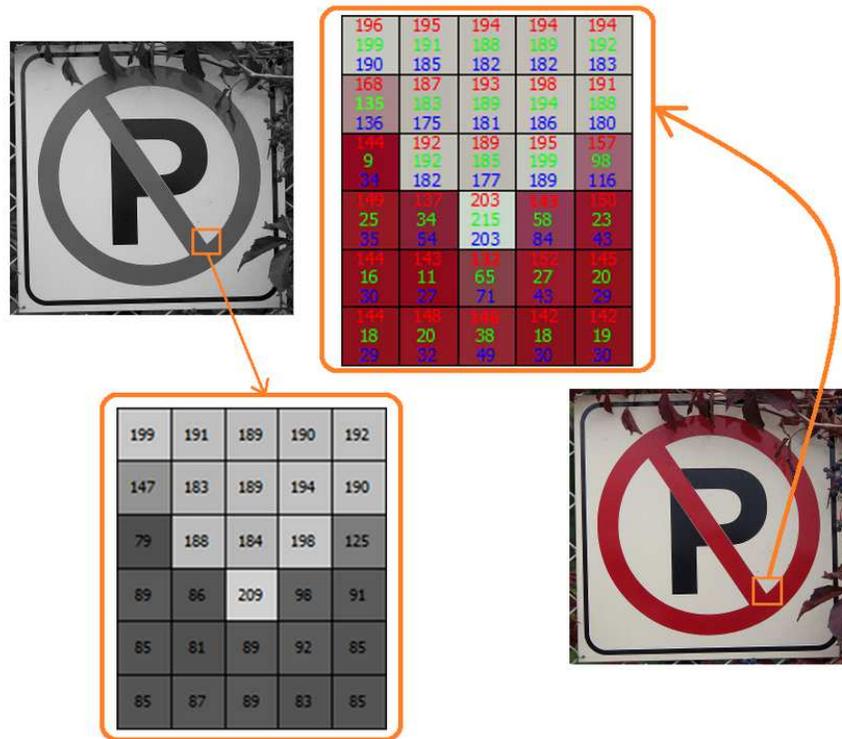


Figura 8. Comparación de imagen en escala de grises contra RGB [29]

En la Figura 8 es posible observar como una imagen en escala de grises es la manera más simple de representación. De la misma manera, el representar una imagen a color implica tener tres canales, por ejemplo: RGB (Red, Green, y Blue), los cuales combinados forman la imagen por completo.

### 2.5.1 Espacios de color

Un espacio de color (también referido como modelo de color) es un método que se utiliza para interpretar, almacenar y reproducir un conjunto de colores.

#### Escala de Grises

En la escala de grises mostrada en la Figura 9 cada pixel está representado con un número entero de 8 bits que corresponde al brillo o a la escala de gris de ese pixel, haciendo posible almacenar 256 diferentes niveles de grises; en esta escala el cero representa al negro absoluto y el 255 al blanco absoluto.



Figura 9. Espacio de color de la escala de grises [29]

## **RGB**

En este espacio de color cada pixel es representado por tres valores enteros de 8 bits, que corresponden a la intensidad del rojo, verde y azul de cada pixel, lo cual permite representar todos los colores combinando distintas cantidades de rojo, verde y azul. Este espacio de color es particularmente conocido por ser utilizado en TVs, LCD y pantallas similares.

La Figura 10 presenta una representación visual del espacio de color RGB y los colores que son posibles de generar combinando los valores de rojo, verde y azul.



Figura 10. Espacio de color RGB [29]

## **HSV (Hue, Saturation, Value)**

En el espacio de color HSV cada pixel está representado por tres valores para: el tono (Hue) o color; la saturación (Saturation) o intensidad del color; y el valor (Value) que representa lo brillante u oscuro que es el color). En la Figura 11 es posible observar la representación del tono del espacio HSV, el cual puede corresponder a un valor entre 0 y 360 grados.

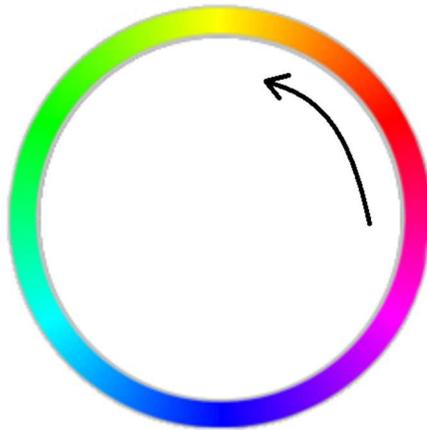


Figura 11. Representación de tono para HSV [29]

### **CIE L\*a\*b\***

El espacio de color CIELAB o CIE L\*a\*b\* especificado por CIE (en Francés, Commission Internationale de l'Éclairage) en 1976 [30] fue desarrollado con la intención de linealizar la representación del color con respecto a la percepción humana para crear un sistema intuitivo. Los colores son representados en tres valores en donde L\* representa la luminosidad, y los componentes a\* y b\* representan el tono y la saturación a lo largo de los ejes verde-rojo y azul-amarillo, respectivamente. La Figura 12 presenta una representación del espectro RGB transformado al espacio de color CIE L\*a\*b\*.

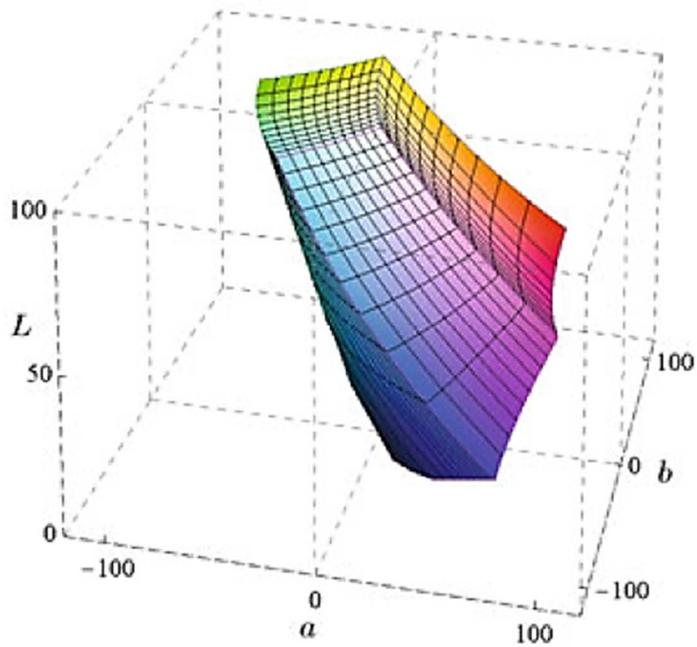


Figura 12. Representación del espacio CIE  $L^*a^*b^*$  [30]

### 2.5.2 Filtros espaciales

En el filtrado espacial el valor del pixel depende de sí mismo y de los pixeles vecinos, por lo tanto, el filtrado espacial es un procedimiento donde el valor de cualquier pixel en la imagen de salida se calcula aplicando algún algoritmo a los valores de los pixeles vecinos.

#### Filtro espacial lineal

Un filtro lineal espacial realiza una operación de suma de productos entre una imagen y un kernel (en español, núcleo); el kernel es un arreglo cuyo tamaño define los vecinos de la operación y cuyos coeficientes determinan la naturaleza del filtro.

La Figura 13 ilustra un filtro espacial lineal usando un kernel de  $3 \times 3$ , en cualquier punto  $(x, y)$  de la imagen la respuesta es la suma de los productos de los coeficientes del kernel y los pixeles comprendidos por el kernel.

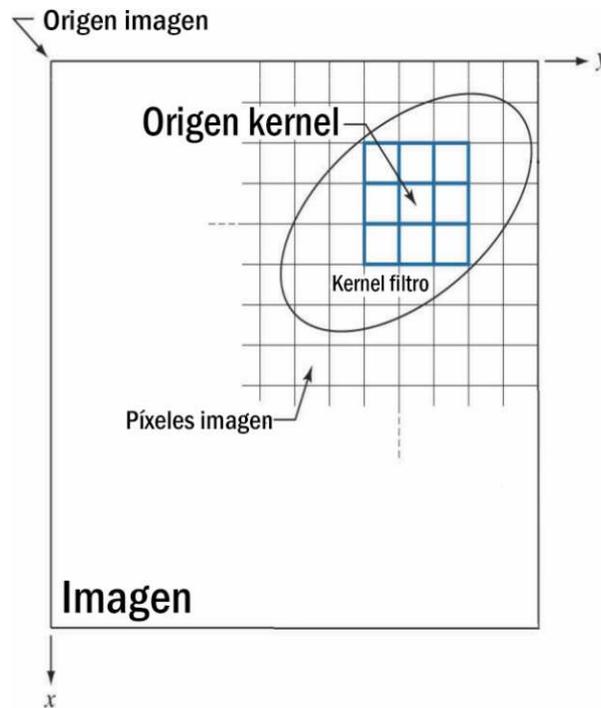


Figura 13. Ejemplo de un kernel de 3 x 3 [31]

### Ruido en una imagen

Las principales fuentes de ruido en las imágenes digitales surgen durante la adquisición o transmisión de las mismas. El rendimiento de una imagen se ve afectada por distintos factores ambientales durante la adquisición, por ejemplo, "al tomar una imagen con una cámara CCD los niveles de luz y la temperatura son factores que afectan la cantidad de ruido en la imagen resultante" [31]. También, durante la transmisión de una imagen a través de una red inalámbrica, esta puede ser dañada por perturbaciones atmosféricas [31].

Dependiendo del tipo de ruido se pueden utilizar diferentes métodos para corregirlo; el moteado, es aparentemente aleatorio y se puede eliminar con un filtro de rango o de mediana; el desenfoque de movimiento puede ser causado por un movimiento uniforme o no uniforme de la cámara, es un problema complejo y existen varios métodos para su eliminación.

En la Figura 14 se ilustra un ejemplo de una imagen con ruido moteado y su eliminación por medio de un filtro de mediana.



(a)



(b)

Figura 14. (a) imagen con ruido; (b) imagen con un filtrado de mediana [32]

### Filtro de suavizado

Para reducir el ruido de una imagen y para realizar un desenfoque se utilizan comúnmente filtros de suavizado, los cuales permiten eliminar información no importante de una imagen, conectar pequeños saltos en curvas o líneas y reducir el ruido de una imagen.

Un filtro lineal de suavizado comúnmente se le llama filtro medio o filtro de mediana, ya que implica sustituir el valor de cada pixel en una imagen con la media de los pixeles vecinos definidos por el kernel.

La Figura 15 presenta el resultado de aplicar un filtro de suavizado utilizando diferentes tamaños de kernel.



Figura 15. Salida del filtro lineal de salida [32]

## Filtro de afilado

Para mejorar los detalles finos o resaltar los detalles borrosos de una imagen se utiliza un filtro de afilado; este se puede realizar mediante el uso de derivadas espaciales, que se pueden aplicar en áreas planas o regiones constantes o a lo largo de discontinuidades o rampas de nivel en escala de grises, siendo las discontinuidades líneas, puntos de ruido o bordes.

La Figura 16 presenta el resultado de aplicar un filtro de afilado a una imagen, resaltando los bordes de la misma.



Figura 16. (a) imagen original; (b) imagen afilada [32]

### 2.5.3 Operaciones morfológicas

Los métodos morfológicos que se utilizan en Álgebra también pueden ser utilizados en las imágenes y pueden ser aplicados a imágenes binarias, con escala de grises y a color. Los operadores morfológicos no son lineales, pero resultan muy intuitivos ya que sus efectos son visibles y fáciles de comprender. Los métodos morfológicos son adecuados para diversas tareas para la evaluación de imágenes, por ejemplo, pre procesamiento, reducción de ruido, extracción y detección de bordes [33].

La Tabla 2 presenta una comparación de algunas operaciones morfológicas en imágenes binarias y en imágenes a escala de grises y el resultado que se genera [32].

Tabla 2. Comparación de operaciones morfológicas aplicadas en imágenes binarias y en imágenes a escala de grises

Operación	Imagen binaria	Imagen a escala de grises
Erosión (en inglés, Erosion)	Reducir.	Oscurecer.
Dilatación (en inglés, Dilation)	Rellenar huecos.	Aclarar.
Apertura (en inglés, Opening)	Abrir espacios entre regiones conectadas.	Reducción de los detalles brillantes
Cierre (en inglés, Closing)	Conectar o rellenar agujeros en regiones.	Reducción de los detalles oscuros.

Las últimas dos operaciones de la Tabla 2 se componen de una combinación de las primeras dos operaciones. La Figura 17 presenta un ejemplo de ello.

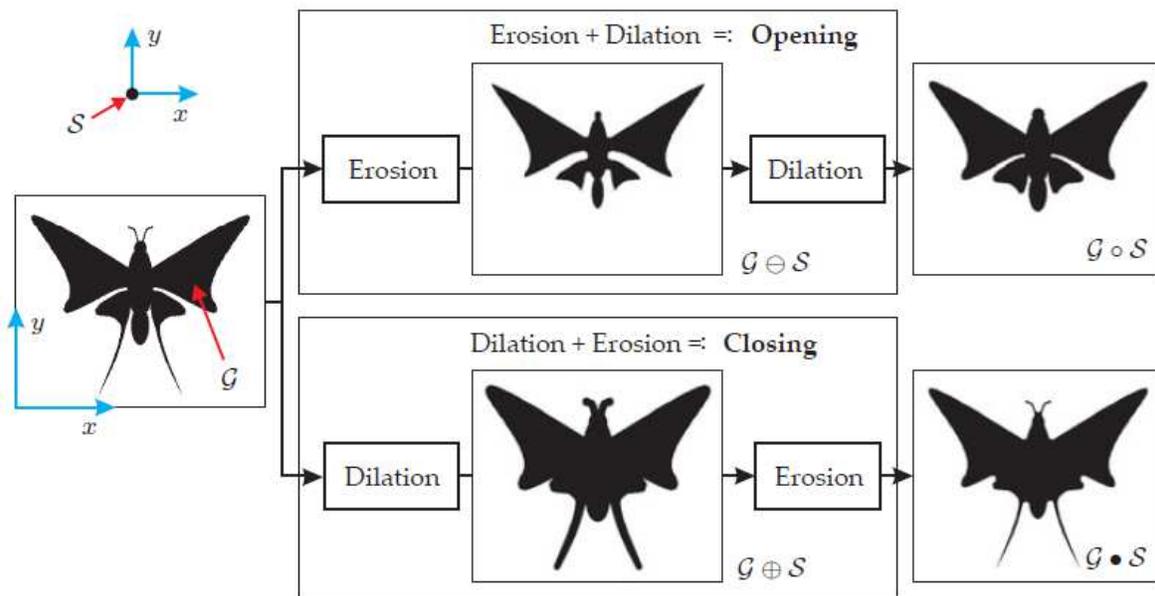


Figura 17. Operaciones morfológicas y su combinación [33]

## 2.5.4 Ecuación de histograma

El brillo y el contraste pueden mejorar la apariencia de una imagen. Para realizar este ajuste de manera automática una solución es obtener el histograma de los canales individuales de la imagen, y a partir de él obtener estadísticas relevantes, como los valores de intensidad mínima, máxima y media, para posteriormente realizar una ecualización de histograma.

La ecualización de histograma distribuye los valores de los píxeles entre un máximo y un mínimo utilizando una función de reasignación (ecualización) de contraste, con el objetivo de crear un histograma con una distribución uniforme. Este método funciona bien para las imágenes en escala de grises, pero las imágenes en color deben ecualizarse en el canal de la intensidad, por ejemplo, en el canal HSV.

La Figura 18 presenta del lado izquierdo una imagen y su histograma, y del lado derecho la imagen y el histograma después de realizar el ecualizado; en el histograma de la imagen de la derecha la distribución de los valores se ha repartido a lo largo de los valores, por lo cual, las características de la imagen se han visto modificadas.

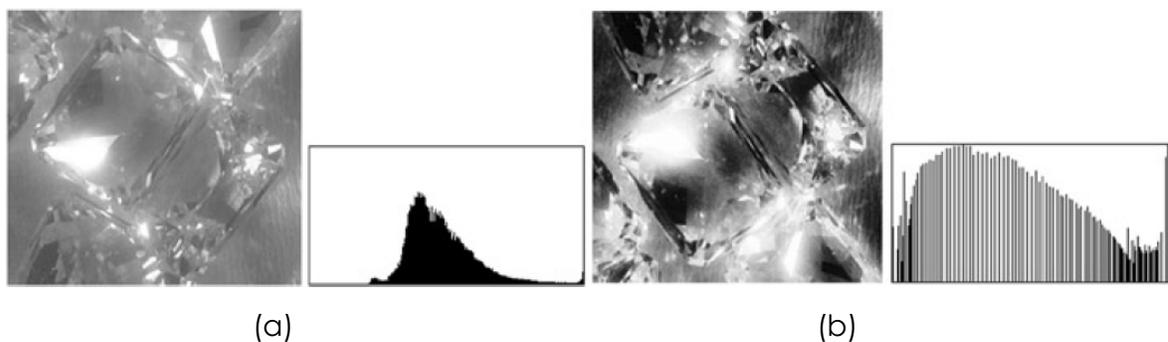


Figura 18. (a) imagen original y su histograma; (b) imagen e histograma ecualizados [34]

## 2.5.5 Segmentación de texto en imágenes

La segmentación del texto en las imágenes permite separar caracteres, palabras, oraciones completas o líneas de texto, dependiendo del detalle a buscar. En [35] se presentan tres niveles de segmentación comúnmente utilizadas:

- Segmentación por línea. Consiste en realizar una búsqueda horizontal en la imagen que compruebe la intensidad de los píxeles y dependiendo del valor, éstos se van agrupando en múltiples regiones, las cuales representan contenidos diferentes en la imagen, pero siendo susceptible a la inclinación entre los mismos caracteres del texto.
- Segmentación por palabra. Consiste en la búsqueda vertical y horizontal en la intensidad de los píxeles, de izquierda a derecha y de arriba abajo, agrupándolos de manera similar al procedimiento de segmentación por línea. La segmentación por palabra permite la corrección de la inclinación de estas, determinando el ángulo de la imagen y girándolo en dirección opuesta.
- Segmentación por carácter. Este procedimiento es similar a la segmentación por palabra, teniendo cuidado en algunas letras, por ejemplo, la h se puede extraer como una l e i. Estos errores son indeseables, resultando que la segmentación sea más complicada en textos con fuentes cursivas.

Existen diferentes técnicas y métodos para realizar la segmentación en los tres niveles anteriormente mencionados. A continuación, se mencionan algunas de estas.

- Conteo de píxeles. Esta técnica necesita de una imagen binarizada y consiste en realizar un escaneo de izquierda a derecha y de arriba abajo; se compara el valor de 0 el cual representa negro y 1 el cual representa blanco. Esta técnica es sencilla de implementar, pero no puede ser utilizada en textos con mucha inclinación, tampoco cuando los caracteres se superponen unos a otros y cuando existe un espacio irregular entre las líneas de texto [22] y [27].
- Histograma de proyección vertical. En este método se obtiene el histograma vertical de la imagen, en el cual las líneas con texto corresponden a picos en el histograma. Por tanto, los espacios vacíos entre picos representan posibles regiones entre diferentes líneas de texto. En [36] se utiliza este método para generar una máscara con la cual se separan los caracteres de la imagen.

### **2.5.6 Umbralización**

El umbralizado es un procedimiento de transformar una imagen en escala de grises a una binaria o en una imagen con distintos niveles de grises utilizando un valor de umbral.

Su meta radica en extraer algunos pixeles de la imagen mientras que otros son removidos. El umbralizado contiene dos regiones de valores en base a un valor de umbral, la región sobre el umbral o la región bajo el umbral, realizando distintas operaciones con cada una de las regiones. El umbralizado puede ser global o local, el global fija un valor de umbral basándose en toda la imagen y el local o adaptativo se utiliza cuando la imagen tiene una iluminación desigual, lo que dificulta la segmentación con un solo umbral.

La Figura 19 presenta de entrada una imagen con iluminación desigual y el resultado de aplicar un umbralizado global contra uno local.

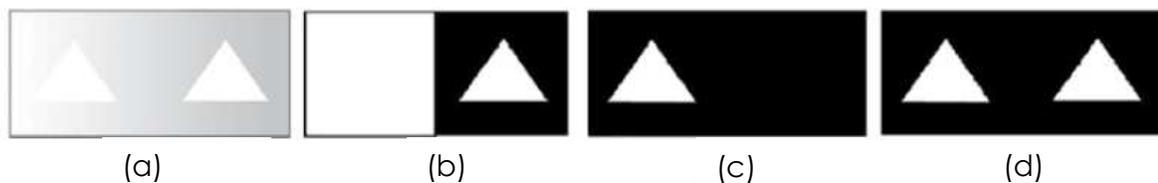


Figura 19. (a) imagen con iluminación desigual; (b) y (c) umbralizado global; (d) umbralizado local [32]

### 2.5.7 Reconstrucción morfológica por dilatación

La reconstrucción morfológica por dilatación de una imagen máscara  $g$ , a partir de una imagen base o semilla  $f$ , está definida como la dilatación geodésica [37] de  $f$  con respecto a la iteración de  $g$  hasta la estabilidad y esta denotada por la Ecuación 1:

$$R_g^\delta(f) = \delta_g^i(f) \quad (1)$$

En donde  $i$  es tal que  $\delta_g^i(f) = \delta_g^{(i+1)}(f)$ .

$\delta_g^i(f)$  representa una dilatación geodésica [37] de tamaño  $i$  con respecto a la imagen semilla  $g$ ; esta operación está definida como el mínimo puntual entre la imagen  $f$  y la operación de dilatación.

La reconstrucción por dilatación de una señal está representada en la Figura 20. En esta figura la estabilidad es alcanzada después de la quinta dilatación geodésica.

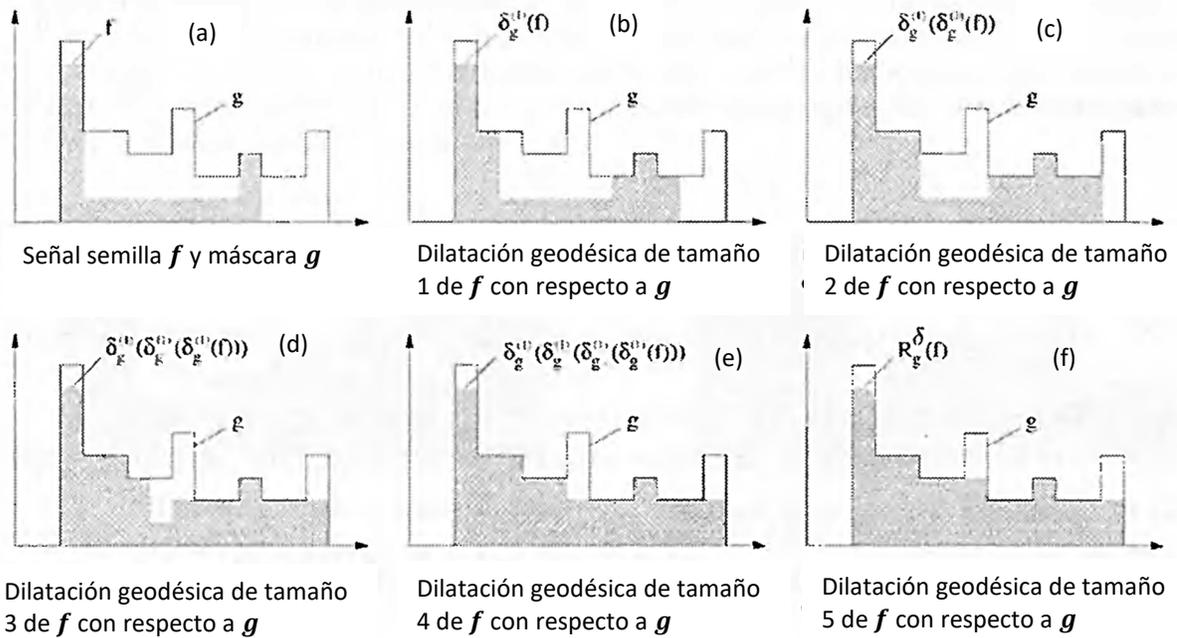


Figura 20. Reconstrucción morfológica por dilatación de tamaño cinco de una señal  $g$  con respecto a una semilla  $f$  [37]

En la Figura 20(f), la dilatación geodésica de tamaño 5 de la semilla  $f$  con respecto a la señal  $g$  es el equivalente a la reconstrucción de  $g$  desde  $f$ ; más dilataciones geodésicas no modificarán el resultado. Esto debido a que la reconstrucción morfológica por dilatación de una imagen máscara  $g$  utilizando una semilla  $f$  es una transformación incremental ( $g_1 \leq g_2 \Rightarrow R_{g_1}^\delta(f) \leq R_{g_2}^\delta(f)$ ), limitada ( $R_g^\delta(f) \leq g$ ) e idempotente ( $R_{R_g^\delta(f)}^\delta(f) = R_g^\delta(f)$ ) [37].

## 2.6 RECONOCIMIENTO DE PATRONES

Los humanos poseen las capacidades de reconocimiento de patrones más sofisticadas en el mundo biológico conocido [31]. Por el contrario, las capacidades de las máquinas se ven superadas en comparación con las tareas que los humanos realizan de manera rutinaria, pero las máquinas de reconocimiento tienen un papel importante en la vida cotidiana. Por ejemplo, existen máquinas que leen códigos de barras, procesan cheques bancarios, inspeccionan calidad en los productos fabricados, leen huellas digitales, clasifican el correo y reconocen el habla.

### **2.6.1 Reconocimiento de texto**

La segmentación de documentos es un problema que ha evolucionado desde su concepción, pero en los últimos años las investigaciones en esta área se han vuelto más complejas; inicialmente el problema era reconocer palabras y caracteres en una línea de texto, posteriormente se aumentó a reconocer todas las líneas de texto en un documento impreso, pero al volverse los documentos más complejos, el problema se migro a identificar las regiones de un documento las cuales contienen texto.

Aplicaciones modernas permiten reconocer texto escrito a mano e impreso coexistiendo en un solo documento, lo cual complica la segmentación y el reconocimiento del texto, debido a la diversidad de estilos de escritura comparados con los textos impresos. Otras aplicaciones que utilizan segmentación de texto y OCR son los motores de búsqueda para buscar documentos escaneados, sobre todo en documentos históricos los cuales tomarían mucho tiempo en transcribirse [38].

### **2.6.2 Reconocimiento óptico de caracteres (OCR)**

OCR (en inglés, optical character recognition) es una tecnología que se utiliza para convertir las letras o palabras escritas o impresas a un formato digital. Comúnmente, se compone de varias etapas: el pre procesado, la segmentación, el pos procesado y la extracción de características.

La Figura 21 presenta las diferentes áreas en las que se puede clasificar el OCR.

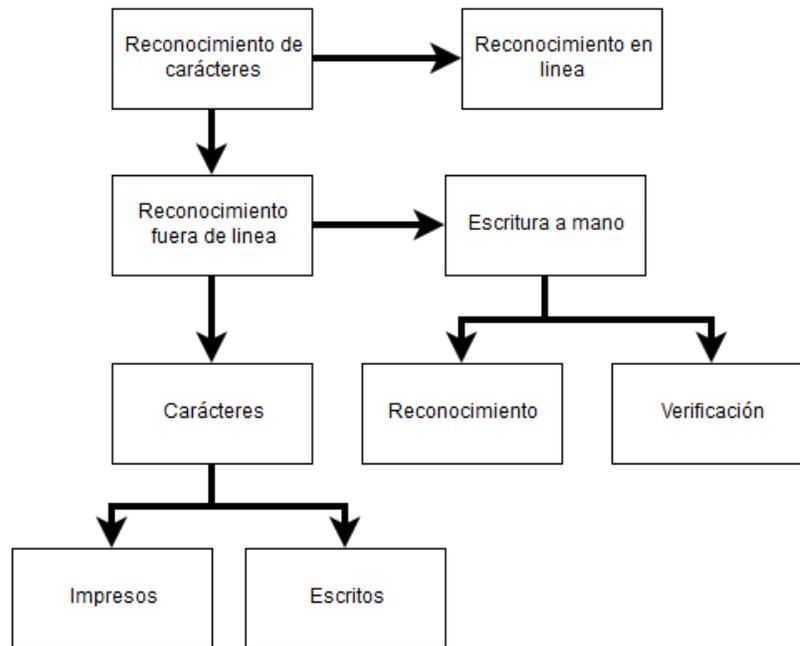


Figura 21. Las diferentes áreas del OCR [39]

El concepto principal en el reconocimiento automático de patrones consiste en primero enseñar a la máquina qué clase de patrones ocurrirán. En OCR los patrones son letras, números y algunos símbolos especiales, la enseñanza de la máquina se realiza mostrando ejemplos de caracteres de todas las clases. Con base en estos ejemplos, la máquina construye un prototipo o una descripción de cada clase de caracteres. Durante el reconocimiento, los caracteres desconocidos se comparan con las descripciones obtenidas anteriormente y se asignan a la clase que ofrece la mayor coincidencia.

Un sistema típico de OCR consiste de diferentes pasos, el primer paso es digitalizar el documento o texto. Cuando se encuentran las regiones que contienen el texto, cada símbolo se extrae a través de una segmentación. Los símbolos extraídos son pre procesados para remover el ruido y facilitar la extracción de características. La identidad de cada símbolo se encuentra al comparar las características extraídas con los descriptores de clases obtenidas a través de un aprendizaje previo. Finalmente, la información contextual se utiliza para reconstruir palabras y números del texto original [39].

### 2.6.3 Tesseract-OCR

Tesseract [40] es un motor de OCR, desarrollado por HP entre 1984 y 1994 y liberado en 2005 para convertirse en código abierto, mantenido actualmente por Google.

HP desarrolló de manera independiente tecnología para el análisis del diseño de la página, Tesseract nunca necesitó su propio análisis del diseño de la página. Por lo tanto, el motor supone que su entrada es una imagen binaria con regiones poligonales de texto definidas. A continuación, se explican los componentes base de motor de Tesseract.

#### **Generación de blobs**

Se realiza un análisis de componentes conectados en el que se almacenan los contornos de los componentes, el cual, a pesar de ser computacionalmente costoso, le permite a Tesseract manejar texto en blanco sobre negro, los contornos se agrupan en blobs (regiones) a través de un filtrado de altura porcentual eliminando las mayúsculas y los caracteres que se tocan verticalmente; la altura media se aproxima al tamaño del texto en la región, permitiendo de manera segura filtrar blobs que son más pequeños que la altura media de los otros caracteres, comúnmente siendo estos blobs pequeños, la puntuación, los signos diacríticos y ruido.

#### **Líneas de texto**

Una vez que los blobs se han filtrado se organizan en líneas de texto y se utiliza una mediana mínima de ajuste de cuadrados para estimar las líneas base del texto; las líneas y las regiones de texto se analizan para un tono o un texto proporcional. Las líneas base se ajustan con mayor precisión utilizando un spline cúbico[41], el cual se ajusta a la partición más poblada mediante un ajuste de mínimos cuadrados [42]. El spline cúbico tiene la ventaja de que es razonablemente estable para este tipo de cálculos, pero pueden surgir discontinuidades cuando se requieren múltiples segmentos de spline. La Figura 22 presenta un ejemplo de una línea de texto con una línea base ajustada de color azul, una línea descendida de color verde, una línea media de color violeta y una línea ascendida de color cian, todas las líneas son paralelas. La línea ascendida de color cian y la línea negra encima de ella aparentan ser rectas; realizando una inspección minuciosa, la línea cian está curvada en relación con la línea negra que en realidad si es recta.

Volume 69, pages 872-879,

Figura 22. Ejemplo de una línea de texto con base curva, la línea azul representa la línea base ajustada, la línea verde representa la línea descendiente, la línea violeta representa la línea media, la línea cian representa una línea ascendida, la línea negra corresponde una línea recta que permite observar la pequeña curvatura en la línea cian [43]

### División de las líneas de texto

Las líneas de texto se dividen de manera diferente dependiendo del espacio entre los caracteres. Tesseract prueba las líneas de texto para determinar si mantienen un tono fijo, cuando el texto tiene un tono fijo divide las palabras en caracteres usando el tono. La Figura 23 presenta el ejemplo de una palabra separada por tono fijo; el espaciado de texto con un paso no fijo o proporcional es una tarea no trivial. Tesseract resuelve este problema midiendo huecos en un rango vertical limitado entre la línea base y la línea media del texto. Los espacios que están cerca del umbral límite se hacen borrosos de modo que se puede tomar una decisión final después de la etapa de reconocimiento de palabras.

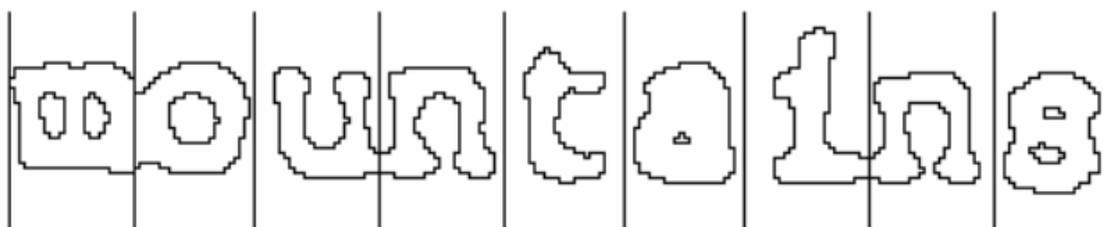


Figura 23. Palabra cortada por tono fijo [43]

### Caracteres unidos

Cuando una palabra resulta insatisfactoria, Tesseract intenta mejorar el resultado cortando el blob con la peor confianza del clasificador de caracteres. Los puntos de corte candidatos se encuentran a partir de vértices cóncavos de una aproximación poligonal [44] del contorno y pueden tener otro vértice cóncavo opuesto o un segmento de línea. La Figura 24 presenta un conjunto de candidatos de puntos de corte con

flechas y el corte seleccionado como una línea a través del contorno donde la r toca la m. Los cortes se ejecutan con base en la prioridad, cualquier corte que no mejore la confianza del resultado se deshace, pero no se descarta por completo por si es necesario reutilizarlo más tarde.

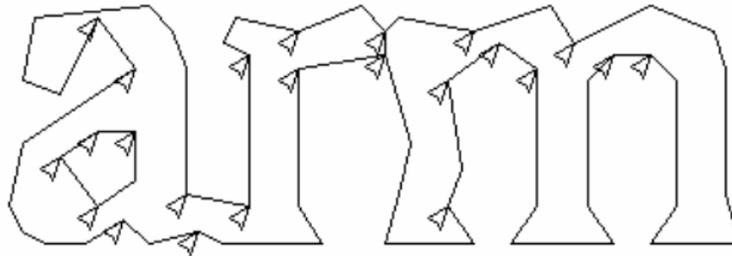


Figura 24. Puntos de corte y líneas de corte [43]

### Caracteres rotos

Cuando las posibilidades de corte se terminan y la palabra no es lo suficientemente satisfactoria, es entregada a un asociador. El asociador realiza una búsqueda a priori de posibles combinaciones de los blobs para convertirse en caracteres candidatos. Este proceso se realiza sin una reconstrucción gráfica, sino que se mantiene una tabla de estados visitados. La búsqueda continúa sacando nuevos candidatos de una cola prioritaria y evaluándolos, clasificando combinaciones no clasificadas de fragmentos. La Figura 25 presenta un ejemplo de caracteres rotos, la cual puede ser reconocida con el procedimiento anterior.

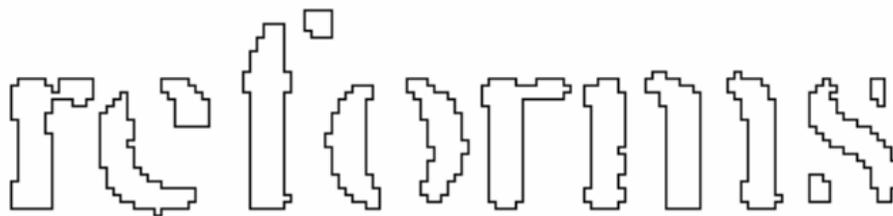


Figura 25. Palabra con caracteres rotos [43]

### Clasificación

En la clasificación de los caracteres se realiza un proceso de dos pasos. En el primer paso, se crea una lista corta de clases de caracteres que podrían coincidir con los desconocidos, cada característica se obtiene de una tabla de búsqueda tridimensional cuantificada de forma aproximada, un vector de bits de clases con las que podría

coincidir y los vectores de bits se suman sobre todas las características. Las clases con los recuentos más altos se convierten en la lista corta. En el segundo paso, cada característica desconocida busca un vector de bits prototipos de la clase dada con los que podría coincidir y luego se calcula la similitud entre ellos. Cada clase de caracteres prototipos está representada por una expresión lógica de una suma de un producto con cada término llamado configuración; por lo que el proceso de cálculo de distancia mantiene un registro de la similitud total de cada característica en cada configuración, así como de cada prototipo. La mejor distancia combinada, que se calcula a partir de la característica sumada y las pruebas de prototipo, es la mejor sobre todas las configuraciones almacenadas de la clase.

### **Datos de entrenamiento del clasificador**

El clasificador fue entrenado con 20 muestras de 94 caracteres de 8 fuentes en un solo tamaño, pero con 4 atributos, normal, cursiva, negrita y negrita cursiva, lo que hace un total de 60160 muestras de entrenamiento [43].

### **Análisis lingüístico**

Tesseract contiene relativamente poco análisis lingüístico. Cada vez que el módulo de reconocimiento de palabras considera una segmentación nueva, el módulo lingüístico elige la mejor cadena de palabras disponible para cada una de las siguientes categorías: palabra más frecuente, palabra de diccionario más frecuente, palabra numérica más frecuente, palabra mayúscula más frecuente, palabra minúscula más frecuente y palabra mejor elegida por el clasificador. La decisión final para una segmentación es simplemente la palabra con la clasificación de distancia total más baja, donde cada una de las categorías anteriores se multiplica por una constante diferente.

Las palabras de diferentes segmentaciones pueden estar conformadas por un número diferente de caracteres. Tesseract genera dos números para cada clasificación de caracteres. El primero, llamado confianza, es menor a la distancia normalizada del prototipo. Esto permite generar una confianza en el sentido de que los números mayores son mejores, pero a cierta distancia. Cuanto más lejos del cero, mayor es la distancia. La segunda salida, llamada puntuación, multiplica la distancia normalizada del prototipo por la longitud total del contorno en el carácter desconocido. Las puntuaciones de los

caracteres dentro de una palabra se pueden sumar, ya que la longitud total del esquema para todos los caracteres dentro de una palabra siempre es la misma.

### **Clasificador adaptativo**

Cada palabra que es satisfactoria se pasa como datos de entrenamiento a un clasificador adaptativo el cual utiliza normalización isotrópica de la línea de base [43]. La normalización de la línea de base facilita la distinción entre mayúsculas y minúsculas, además de mejorar la inmunidad al ruido. El principal beneficio de la normalización es la eliminación de la relación de aspecto de la fuente y cierto grado de ancho de trazo de fuente; también hace que el reconocimiento de subíndices y superíndices sea más simple, pero requiere una función de clasificador adicional para distinguir algunos caracteres en mayúsculas y minúsculas. El clasificador tiene la oportunidad de reconocer con mayor precisión el texto posterior.

Dado que el clasificador adaptativo puede aprender algo útil demasiado tarde para poder contribuir a la detección de la primera información, se ejecuta una segunda etapa de análisis sobre el texto, en el que las palabras que no se reconocieron anteriormente se reconocen nuevamente.

### **Fase final y obtención del texto**

Se resuelven los espacios borrosos y se verifican las hipótesis alternativas para poder ubicar texto con letras pequeñas, obteniendo finalmente una salida de texto digital.

La Figura 26 presenta las etapas del motor de manera resumida.

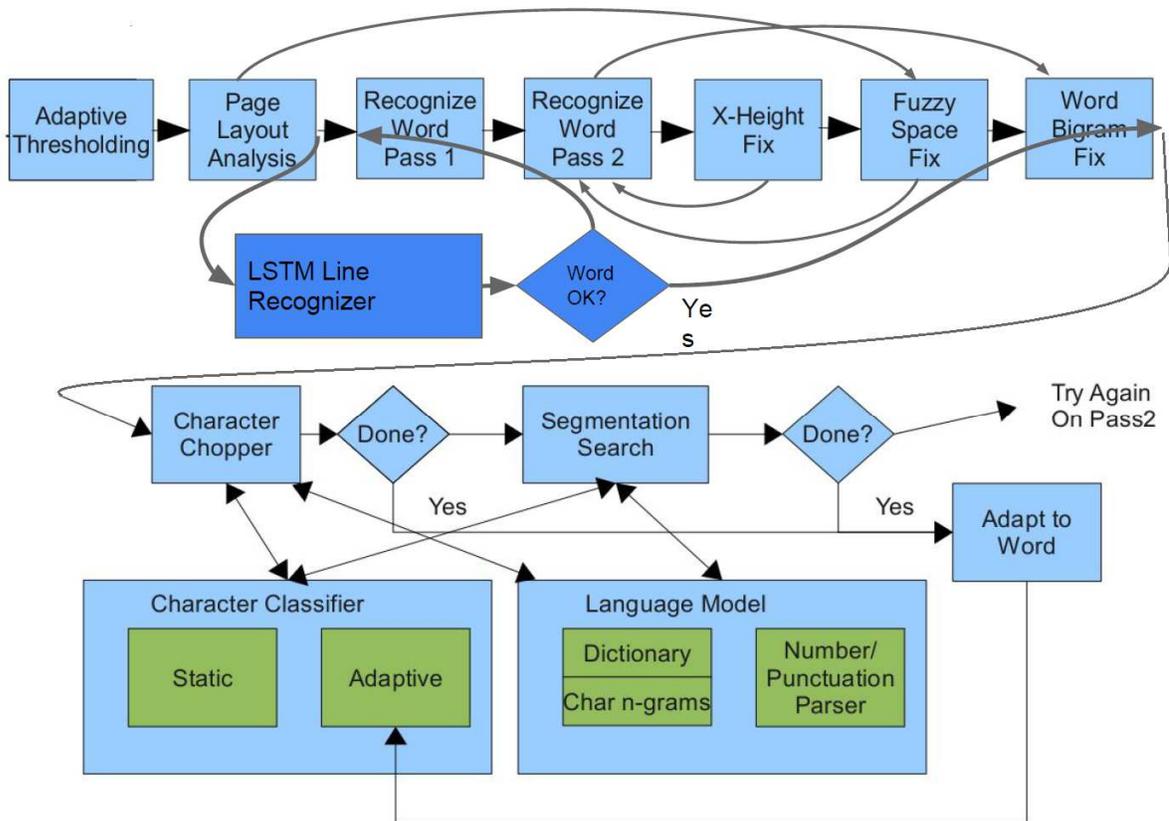


Figura 26. Etapas del motor de OCR Tesseract[43]

Tesseract en su versión 3.00 implementó nuevos idiomas, incluyendo chino, japonés y coreano, en su versión 3.02 implementó soporte para texto bidireccional y la habilidad de reconocer múltiples idiomas en una sola imagen, en su versión 4.00 se agrega un nuevo motor de OCR basado en redes neuronales LSTM [45] que se centra en el reconocimiento de líneas, pero que también admite el motor antiguo utilizado en Tesseract 3, mediante el reconocimiento de patrones en caracteres [40].

En el presente trabajo se utiliza la versión 4.0 de Tesseract junto con los paquetes de idiomas de español e inglés, ya que al utilizar un motor basado en redes neuronales LSTM, se espera obtener un mejor resultado que con el motor anterior de OCR.

## 2.7 CLUSTERIZACIÓN

### 2.7.1 K- medias [46]

K-means [46] (en español, K medias) “es un método de agrupamiento cuyo objetivo es agrupar un conjunto  $n$  de datos en  $k$  grupos, en el que cada observación pertenece al grupo cuyo valor medio o centroide es más cercano” [46].

Dado un conjunto de observaciones  $(x_1, x_2, \dots, x_n)$  donde cada observación es un vector real de  $d$  dimensiones, k-medias construye una división de las observaciones en  $k$  ( $k \leq n$ ) conjuntos, a fin de minimizar el error de la suma de los cuadrados dentro de cada grupo. El método K-means se define como [46]:

$$\operatorname{argmin}_{SSE} \quad \text{donde } SSE = \sum_{i=1}^n \sum_{j=1}^k w^{(i,j)} \|x^{(i)} - \mu^{(j)}\|^2. \quad (2)$$

donde

$\mu^{(j)}$  – cluster representativo o centroide para el cluster  $j$

$w^{(i,j)} = 1$  si la muestra  $x^{(i)}$  se encuentra en el cluster  $j$ , de lo contrario  $w^{(i,j)} = 0$

$SSE$  – suma de la estimación cuadrática del error

## 2.8 MÉTRICAS DE EVALUACIÓN

Las métricas de evaluación permiten conocer el desempeño del sistema al permitir la comparación de la predicción del sistema contra un ground truth (en español, base verdadera).

### 2.8.1 Clasificación binaria

En una clasificación binaria, en la cual la respuesta es sí o no, cada una de las predicciones tiene cuatro diferentes resultados. El verdadero positivo (TP, del inglés true positive) y el verdadero negativo (TN, del inglés true negative) son clasificaciones correctas; un falso positivo (FP, del inglés false positive) ocurre cuando el resultado es predicho incorrectamente como un sí o positivo, cuando realmente es un no o negativo; un falso negativo (FN, del inglés false negative) ocurre cuando el resultado es predicho incorrectamente como negativo cuando realmente es positivo.

La tasa de verdaderos positivos se calcula dividiendo TP entre el número total de positivos, el cual está formado por TP + FN; la tasa de falsos positivos se calcula dividiendo FP por el número total de negativos, FP+TN.

La tasa de éxito general es el número de clasificaciones correctas dividido por el número total de clasificaciones y, finalmente, la tasa de error es uno menos el resultado anterior [47].

### **2.8.2 Clasificación multiclase**

En una clasificación multiclase los resultados comúnmente son presentados en una matriz de confusión de dos dimensiones, con una columna y una fila para cada clase, cada elemento de la matriz muestra el número de ejemplos de prueba para los cuales la clase real es la fila y la clase predicha es la columna, los buenos resultados corresponden a grandes números en la diagonal principal y elementos pequeños, idealmente cero, fuera de la diagonal.

### **2.8.3 Precisión y exhaustividad**

“La precisión (precision en inglés) se puede definir como el número de instancias recuperadas que son relevantes y la exhaustividad (recall en inglés) es la fracción de instancias relevantes que han sido recuperadas” [48].

### **2.8.4 F-Measure**

La medida F es una métrica que combina la exhaustividad y la precisión, y está dada por la Ecuación 3:

$$F - Measure = \frac{2 \times recall \times precision}{recall + precision} = \frac{2 * TP}{2 * TP + FP + FN} \quad (3)$$

## **CAPÍTULO 3. ESTADO DEL ARTE EN SISTEMAS DE RECONOCIMIENTO DE TEXTO**

En este proyecto el reconocimiento de texto es esencial para la verificación de las etiquetas en los conductores recubiertos. En este capítulo se presenta una descripción y análisis del trabajo relacionado relevante en este tema.

### **3.1 OPTIMIZACIÓN DE EL PRE PROCESADO DE LA IMAGEN CAPTURADA**

En Manwatkar y Singh [49] la parte de pre procesado se realiza para remover ruido de la imagen creado por el escaneo de los documentos. En Chen, Odobez y Bourlard 2004 [50] presentan diferentes técnicas para eliminar la variación en el tamaño de los caracteres y la variación en los tonos de grises en el espacio de características antes del entrenamiento del algoritmo, dividiéndose en dos partes; en la primera parte, el esquema de localización extrae rápidamente los bloques de texto en las imágenes con una tasa de rechazo baja del 0.24%, permitiendo extraer más líneas de texto individuales y normalizar el tamaño del texto; en la segunda parte, se realiza una verificación en un conjunto de espacios característicos los cuales son invariables a los cambios de escala de grises.

En Choudhary, Rishi y Ahlawat 2013 [51] se usaron técnicas de pre procesado similares a los trabajos anteriores. Sin embargo, en éste se observó que la técnica de adelgazamiento y esqueletización permitió disminuir el grosor del texto a contornos delgados y uniformes. En Rodríguez Rodríguez et al. 2016 [52] cada pixel en el área de interés se reemplaza por el mínimo local de su vecindario de Moore, mejorando de esta manera la imagen fragmentada por la impresión de inkjet para posteriormente ser validada.

### **3.2 DETECCIÓN DE PLACAS VEHICULARES**

La detección de placas vehiculares de manera automática en tiempo real es un problema que presenta distintas dificultades al recuperar la imagen de una placa de un vehículo en movimiento, segmentar los caracteres en la imagen y realizar el

reconocimiento óptico de caracteres a través de algoritmos de aprendizaje automático (Machine Learning) y aprendizaje profundo (Deep Learning).

En Hashmi et al. 2019 [53] los autores proponen un sistema de ALPR (del inglés automatic licence plate recognition) en el cual sugieren dividir un video de entrada entre 50 y 80 cuadros por segundo para obtener buenos resultados; se realiza un pre procesamiento para obtener solamente la placa del vehículo de entre todos los otros objetos en la imagen; se obtienen los contornos de la imagen y se eliminan aquellos que no aparentan ser caracteres por su geometría; se agrupan todos los contornos que pertenecen a la misma placa; y se extraen los números de placa de la imagen original. Posteriormente, se realiza la segmentación de los caracteres separando cada uno de los caracteres de las placas removiendo símbolos que no son deseados y manteniendo solamente símbolos de la A a Z y del 0 al 9; se procesan las imágenes de cada uno de los caracteres; y se mejoran las características para asegurar que la imagen de entrada al motor OCR es consistente. Para la parte de OCR se realizaron dos aproximaciones, una con KNN (del inglés k nearest neighbour) y otro con CNN (del inglés convolutional neural network); para KNN, la imagen de entrada se convierte en un vector y se ingresa al clasificador para entrenamiento y pruebas, pero como las imágenes que se obtienen no son similares de cierto modo a las imágenes de entrenamiento, el clasificador KNN tiene dificultad en la predicción. Por otro lado, el clasificador CNN está formado por diferentes capas de neuronas, donde cada neurona recibe una entrada y realiza un producto punto con ciertos filtros, los cuales se utilizan para modificar una imagen de la manera que se desea. La convolución es una operación de vecindad en la que cada pixel de salida es la suma ponderada de los pixeles de entrada vecinos. El clasificador CNN se entrenó usando 32480 imágenes de caracteres, clasificando cada imagen en una de las 36 clases que van desde 0-9 y A-Z. El clasificador KNN obtuvo una exactitud del 68.03% comparado contra el 85% obtenido por el clasificador CNN. Los autores remarcan la importancia de la etapa de pre procesamiento la cual afecta directamente a los clasificadores, permitiendo mejorar los resultados obtenidos inicialmente.

En Farhat et al. 2018 [54] los autores presentan otro sistema de reconocimiento de placas vehiculares al cual llaman sistema ANPR (del inglés, automatic number plate recognition) y proponen cuatro algoritmos para la etapa de OCR. Los algoritmos se basan en la

extracción de características a través de cruce de vectores, división por zonas (zoning en inglés), el resultado de mezclar el cruce de vectores con división por zonas y técnicas de coincidencia de plantillas. El sistema propuesto puede reconocer un carácter en 0.63 ms con una precisión del 99.5%.

En Hendry y Chen 2019 [55] los autores presentan un sistema ALPR pero utilizando un framework de aprendizaje profundo llamado YOLO (del inglés you only look once)-darknet, utilizando 7 de las 27 capas convolucionales de YOLO para detectar una sola clase. El método de detección de las placas utiliza un SWSCD (del inglés, sliding window single class detection) [56, p. 17]. Este clasificador está conformado por 36 etiquetas las cuales son las 25 letras del alfabeto, 10 números y símbolos de la placa. Posterior a la detección de todos los objetos en la imagen, el sistema realiza una localización y agrupamiento de las letras y números en la región de la placa. El objetivo fue el reconocer placas de Taiwán; se utilizó un conjunto de datos AOLP (del inglés, application-oriented license plate) [57], el cual contenía placas vehiculares de 6 dígitos. El sistema obtuvo una exactitud del 98.22% en la detección de las placas y un 78% en el reconocimiento de la placa. El sistema ejecuta una sola fase de detección, la cual toma entre 800ms y 1s por cada imagen que le es enviada. El sistema fue probado con condiciones complejas de reconocimiento como lluvia, obscuridad e imágenes poco concisas.

### **3.3 INSPECCIÓN VISUAL DE PRODUCTOS**

La inspección visual de productos de manera automática en aplicaciones en tiempo real requiere de un sistema que sea robusto a distintas dificultades. Por ejemplo, se requiere que el sistema permita obtener la imagen de un producto que puede estar moviéndose a alta velocidad, verificar la calidad de la impresión y realizar reconocimiento por OCR.

En Pedersen, Nasrollahi y Moeslund 2016 [36] se presenta un término importante dentro de los sistemas de reconocimiento de texto, el cual es el OCV (optical carácter verification). Los autores proponen un sistema de clasificación que evalúa el desempeño de la tarea de impresión utilizando algunas medidas de calidad para cada carácter y símbolos. El propósito del sistema es permitir a los clientes la calificación del texto impreso,

mientras que, para el fabricante de la impresora, sirve de retroalimentación para la mejora y optimización de su sistema de impresión. Para la parte de clasificación, se utilizó un clasificador KNN calculando la distancia entre cada uno de los caracteres y los caracteres de referencia, obteniendo una tasa de reconocimiento del 98.69% y una precisión del 98.57% en el sistema.

En Rodríguez Rodríguez et al. 2016 [52] los autores presentan un sistema para realizar una supervisión visual a muy alta velocidad de codificación impresa industrial en latas de bebidas. Para esto, se realiza una valoración de la imagen óptima para el sistema y se extraen las formas de la imagen; estas son los caracteres que comprenden el código de fecha de vencimiento para posteriormente comparar con el código esperado en la lata. La verificación de las latas se realiza después de su impresión, por lo cual se tiene una tasa de 35 latas por segundo, con condiciones de adquisición de imágenes problemáticas debido a la superficie metálica de la lata y a la iluminación que se necesita para evitar generar reflejos. Una vez obtenida la valoración de la imagen óptima para el sistema, se realiza un pre procesado para poder separar el texto en líneas considerando la dificultad presentada por el interlineado de estas, para finalmente comparar los caracteres obtenidos en las líneas con los esperados. Este sistema logró una precisión del 95%.

En De Sousa Ribeiro et al. 2018 [58] los autores presentan un sistema para la detección de las fechas de expiración en envases de alimentos, considerando como crítico el correcto reconocimiento y codificación de esta fecha para una distribución apropiada de los productos, eliminando el posible riesgo a la salud causado por errores en el etiquetado. Se propuso un sistema basado en redes neuronales profundas incluyendo a nivel global una red neuronal convolucional para la evaluación de la calidad de la imagen, y una red local completamente convolucional (FCN, del inglés fully convolutional network) [59] para la localización de la fecha. Este sistema obtuvo una precisión del 98%.

### **3.4 RECONOCIMIENTO DE TEXTO EN DOCUMENTOS**

El reconocimiento de texto en los documentos presenta un área de estudio en el cual el tipo de documento con el que se trabaje, el origen, la manera de digitalizarlo, el texto

que contiene, ya sea impreso o escrito a mano, presenta diversas problemáticas. Éstas han sido abordadas en los siguientes estudios.

En Bhunia et al. 2018 [60] se presenta primero una selección del canal la cual permite obtener mejores características de la imagen utilizando una máquina de soporte vectorial (SVM, del inglés support vector machine); se realiza una extracción de características para reconocimiento de palabras, combinándolo con un modelo de redes ocultas de Markov. Para evaluar el desempeño del sistema propuesto se utilizaron los conjuntos de prueba ICDAR 2013 [61] y textos Devanagari, obteniendo una precisión del 75.41% y del 72.87%, respectivamente.

En Zhong y Cheriet 2015 [62] los autores presentan un método para la identificación y reconocimiento de texto basado en *tensor representation learning base para image patch análisis* (TRL-IPA); este método puede ser aplicado no solo a imágenes binarizadas, sino también a imágenes en escala de grises y a color. El método se basa en una formulación general del algoritmo de aprendizaje de representación de tensor convergente (CTRL) [62]. Con este método se logró obtener como resultado en la identificación de texto una precisión del 98.52% para texto en chino, 96.11% para árabe y 93.03% para cirílico.

En Chen, Odobez y Boulard 2004 [50] se estudia la detección y reconocimiento de texto en imágenes y videos en situaciones donde el fondo de la imagen es compleja. El método propuesto se divide en dos partes principales, la detección de líneas de texto a través de una máquina de soporte vectorial (SVM) y un perceptrón de múltiples capas (MLP del inglés, multi layer perceptrón) entrenado para remover falsas alarmas en las líneas de texto. El método propuesto incorpora el paso tradicional de segmentación de caracteres seguido de un algoritmo de OCR. Se propuso un algoritmo de restricción de consistencia en escala de grises para mejorar los resultados de la segmentación hasta en un 20%. El sistema produjo una tasa del 97% de reconocimiento de caracteres y una tasa de reconocimiento de palabras de más del 93%.

En Rao 2005 [63] se presenta un enfoque utilizando redes neuronales como base para un algoritmo de reconocimiento de caracteres escritos a mano, inicialmente se presenta una descripción general de diferentes técnicas de reconocimiento de caracteres

existentes. Posteriormente, se desglosan de una manera general los componentes de un sistema OCR, proponiendo un algoritmo basado en redes neuronales y obteniendo una tasa de reconocimiento de 95.44%. En este trabajo, se presentaron algunas aplicaciones de sistemas OCR como: imágenes de facturas, industria legal, bancos, aplicaciones médicas, sistema CAPTCHA, repositorios institucionales y bibliotecas digitales, reconocimiento óptico de música, reconocimiento automático de números y reconocimiento de escritura a mano.

En Amirgaliyev, Kuvatov y Bairbatyr 2017 [64], los autores realizan OCR en tarjetas de identificación de Kazajistán, a través de la implementación de una red neuronal convolucional (CNN) cuyas capas son: entrada, convolución, ReLU, MaxPooling, convolución, ReLU, MaxPooling, convolucion, ReLU, MaxPooling, Linealización, ReLU, Linealización, ReLU, Linealización y LogSoftMax; realizando como etapa de pre procesado un cambio de escala de grises y un ajuste del tamaño de la imagen a un tamaño fijo. Este método logró una precisión del 94.55% en los caracteres kazajos, 96.83% en caracteres en inglés y 99.18% en símbolos de dígitos.

Los enfoques o métodos basados en CNN también se utilizan para mejorar los resultados del OCR. En Wei, Sheikh y Rahman 2018 [65] los autores presentan un enfoque utilizando redes neuronales, específicamente un modelo pre entrenado llamado Inception V3. El sistema se divide básicamente en cuatro partes. Primero se realiza el pre procesado de documentos escaneados, los cuales son segmentados utilizando un algoritmo de máximas regiones extremas estables. Posteriormente, la imagen pasa por Inception V3 utilizando transferencia de aprendizaje para tomar la salida (el valor de los pesos del clasificador) y pasarla por otro modelo de red neuronal para poder, finalmente, tener una clasificación. Para tomar un punto de referencia de su modelo contra los demás modelos de OCR se comparó contra OCR.space [66]; OCR.space es un motor de OCR de código abierto y un servicio de la compañía a9t9 software GmbH [66]. El sistema Inception V3 logró una mejor precisión de reconocimiento en un promedio de 78% para imágenes de baja calidad, reduciendo en 21.5% la tasa de error comparada con OCR.space; en imágenes de buena calidad se mantuvo en un 90.6% de precisión.

### **3.5 SISTEMA CAPTCHA**

En Modi y C. 2017 [67] se presenta el sistema CAPTCHA [68], el cual es un programa que puede generar y calificar pruebas que los humanos pueden aprobar pero que los programas de computadores actuales no pueden; el sistema CAPTCHA genera una imagen que contiene una serie de letras o números oscurecida por diferentes técnicas de distorsión, variando la fuente, el tamaño, el fondo y agregando segmentos aleatorios. Este sistema permite diseñar algoritmos de eliminación de ruido y segmentación robustos para sistemas OCR.

### **3.6 COMBINACIÓN DE LAS DIFERENTES TÉCNICAS**

A continuación, se presentan diferentes combinaciones de métodos de pre procesado, segmentación y clasificación que han sido empleados en el trabajo relacionado relevante para el tema estudiado.

En [69] se realiza segmentación de caracteres a través de métodos heurísticos, adelgazamiento de la imagen, extracción de características basadas en filtros Gabor y clasificación con SVM; el resultado de este sistema les permitió obtener una segmentación correcta del 97.5%, pero al combinarlo con una clasificación por SVM la precisión para el reconocimiento de caracteres sube hasta 99.4%.

En [70] se realiza reducción de ruido, filtros Gaussianos pasa bajos, corrección de inclinación con KSC, normalización del tamaño de la imagen, extracción de características con patrones locales binarios, clasificación utilizando el vecino euclidiano más cercano. Se logró obtener en el sistema una precisión del 96.5% al 96.7%, dependiendo del caso.

En [71] del motor Tesseract, se utiliza la función incluida que permite asignar una caja o recuadro a cada uno de los caracteres en la imagen, adicionado a esto, se re entrenó el motor utilizando diferentes imágenes en las cuales se varió el tamaño y el tipo de fuente del texto contenido en la imagen. Como resultado se obtuvo una precisión del 81% en la asignación de los recuadros a cada uno de los caracteres.

En [72] se realiza binarizado y reducción de ruido para la generación del texto codificado a partir de los caracteres segmentados. En este trabajo se revisaron cuatro métodos: a) cruce de vectores obteniendo una tasa de reconocimiento del 90.43%; b) zoning obteniendo una tasa de reconocimiento del 85.3%; c) la combinación del cruce de vectores con zoning obteniendo una tasa de reconocimiento del 90.43%; y d) la correlación de plantillas obteniendo una tasa de reconocimiento del 99.5%.

En [73] se realiza la detección y corrección de sesgo en el pre procesamiento, la binarización, eliminación de ruido, normalización y el adelgazamiento de las imágenes de entrada. En la extracción de características se obtienen características estructurales dividiendo la imagen en cuatro cuadrantes y características de transformación global obtenidas con la transformada Wavelet, obteniendo 128 características globales y 21 estructurales. En la clasificación se usó una red neuronal y se obtuvo una precisión del 91% para los caracteres y 97.60% para números

En [74] se emplea lógica difusa y características SIFT (del inglés, scale invariant feature extraction). En la primera etapa se construye una pirámide Gaussiana y posteriormente se obtienen los máximos locales. Luego, utilizando SIFT se refinan las características obtenidas, calculando varios parámetros como: el valor de escala local de la característica; el tamaño de la característica en la imagen original; el indicador de si es un borde o no; y el ángulo de orientación del borde local. Las características extraídas en el algoritmo SIFT se convierten en los valores que aparecen en la función de lógica difusa, teniendo cada carácter su propio algoritmo de lógica difusa. Los resultados obtenidos de este algoritmo lograron una precisión del 90.75% en la extracción de los caracteres.

En [75] se pre procesa la imagen y se compara con plantillas en una base de datos; una vez realizada la comparación se encuentran los puntos característicos de las imágenes y las características se extraen del área de interés, logrando una precisión del 92% en el reconocimiento de caracteres.

En [76] se realiza un pre procesado y segmentación de la imagen. En la segmentación las líneas de caracteres se separan y se enumeran ya que resultan esenciales para delimitar los límites de la detección. Posterior a esto, se separan los caracteres y se

extraen las características de cada uno. El siguiente paso consiste en entrenar el algoritmo de aprendizaje basado en perceptrones multi capa, el cual puede ser utilizado para entrenar y clasificar. El sistema logró una precisión del 91.53% para caracteres aislados y una precisión del 80.65% para caracteres en sentencias.

En [77] se realiza template matching (en español, coincidencia de plantillas) y se utiliza un algoritmo back propagation (en español, programación hacia atrás). Template matching se utiliza para extraer las características de la imagen; se compara con una base de datos y se elige el carácter con el valor más alto de coincidencia. El algoritmo back propagation utiliza un mecanismo inverso para encontrar el error y lo reduce propagándolo hacia atrás. El problema de este algoritmo se presenta cuando se tienen letras no identificadas, ya que esto conlleva a un resultado erróneo en el reconocimiento del carácter. El algoritmo permitió obtener una precisión del 91.82% en el reconocimiento de caracteres.

Al combinar varias técnicas de pre procesado, segmentación y clasificación queda claro que, al modificar las diversas técnicas utilizadas en la creación de un algoritmo de OCR, es posible lograr maximizar los mejores resultados. En este trabajo se utilizaron distintas técnicas en las diferentes etapas del algoritmo, buscando con ello mejorar la etapa de pre procesado y evitar errores en la etapa de extracción de errores y ruido.

### **3.7 DISCUSIÓN**

Los trabajos presentados en la Sección 3 proponen diferentes técnicas para la segmentación, clasificación y reconocimiento de texto. Así como la utilización de motores de deep learning pre entrenados como OCR.space, Inception V3 e inclusive Tesseract. También se presentaron enfoques que utilizan redes neuronales convolucionales.

Por lo general, previo al proceso de reconocimiento de texto se realiza un pre procesado de las imágenes, ya que comúnmente estas contienen cierto ruido que puede causar problemas durante el proceso de reconocimiento, como se presentó en la Sección 3.1.

En el t3pico de segmentaci3n de texto, dependiendo de la aplicaci3n, se pueden proponer t3cnicas de segmentaci3n inclusive fuera de las mencionadas en la Secci3n 3.6, por ejemplo: m3todos de agrupamiento (clustering en ingl3s); m3todos de detecci3n de bordes, ya sea por contornos o cambios en el color; m3todos de segmentaci3n orientada a regiones; m3todos multi escala; m3todos de transformaci3n divisoria en base al gradiente; hasta m3todos semi autom3ticos basados en redes neuronales.

La finalidad de la aplicaci3n define el tipo de clasificaci3n o algoritmo que permite realizar la tarea a desarrollar, ya sea detecci3n de placas vehiculares (revise Secci3n 3.2), inspecci3n visual de productos (revise Secci3n 3.3) o reconocimiento de texto en documentos (revise Secci3n 3.4).

En la Tabla 3 se presenta una comparaci3n de las caracter3sticas del trabajo relacionado contra el propuesto en el presente proyecto, comparando las caracter3sticas como espacios de color y la robustez a diferentes situaciones que causan problemas.

Tabla 3. Trabajo relacionado versus trabajo propuesto

Ref.	Espacio de color			Robusto a						Pre procesado		Segmentación							Método de clasificación					
	RGB	LAB	Grises	iluminación	escala	rotación	interlineado	objetos en movimiento	Ruido en la imagen	Imágenes de baja calidad	Etapas de pre procesado	Reconstrucción Morfológica.	Operación Morfológicas	Contornos	Clustering	SVM	POG	DNN	K-means	DNN	Características invariantes	HMM	NN	TRL-IPA
[63]			X				X	X	X	X	X			X						X				
[54]			X	X		X	X	X			X				X						X			
[60]	X	X	X		X		X			X	X				X							X		
[36]			X						X	X	X						X						X	
[58]										X	X							X		X				
[62]	X		X				X	X		X									X					X
Propuesto		X		X	X			X	X		X	X	X	X					X					

En el presente trabajo la detección del texto está enfocada totalmente en el caso del texto impreso en conductores eléctricos, en donde el texto impreso sobre el conductor está formado en su totalidad por diferentes tomas continuas, por lo que es necesario procesar la imagen que es capturada en su momento. Dentro de los problemas que conllevan este análisis se encuentran: la variación entre el color del texto y el cable; el tamaño del texto que varía dependiendo del tamaño del conductor eléctrico y del fabricante.

Para solucionar estas problemáticas, en este trabajo se segmenta el texto de la imagen para posteriormente realizar el reconocimiento del mismo. Se adopta como algoritmo de OCR el motor Tesseract, desarrollado por HP y mantenido por Google [43], por su facilidad de implementación y robustez.

## CAPÍTULO 4. MÉTODO PROPUESTO

Para mejorar la inspección del etiquetado en los conductores eléctricos, la cual se realiza de manera manual en algunas compañías, en esta tesis se propone un sistema de visión inteligente. Este sistema, a partir de una serie de imágenes digitales, permite validar el texto sobre el conductor eléctrico para facilitar la toma de decisiones o alertar al operador sobre un mal funcionamiento en el proceso de etiquetado. El sistema de visión propuesto permitirá reducir la cantidad de conductor mal etiquetado.

### 4.1 GENERALIDADES DEL SISTEMA DE VISIÓN PROPUESTO

La Figura 27 muestra los bloques que componen el esquema general del procesamiento y análisis de las imágenes en el sistema de visión propuesto para la inspección de etiquetas de conductores eléctricos.<sup>7</sup>

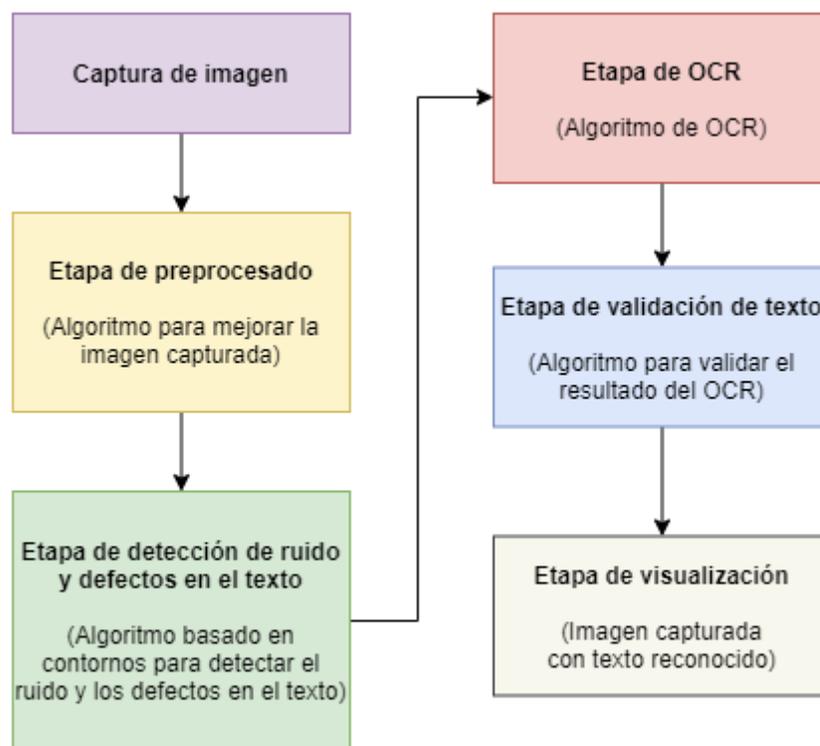


Figura 27. Esquema del procesamiento y análisis de imágenes del sistema de visión para la inspección de etiquetas de conductores eléctricos

Fuente: elaboración propia

El sistema propuesto permite realizar la validación del texto que se imprime en el conductor eléctrico que se está fabricando mediante una inspección visual. El funcionamiento general del sistema propuesto es el siguiente:

- Como primer paso se captura la imagen del conductor eléctrico con el sistema de adquisición de imágenes desarrollado en la empresa.
- Una vez capturada la imagen del conductor se realiza un pre procesamiento, el cual permite extraer información propia o característica de la imagen para mejorar su calidad para ingresarla al algoritmo de OCR.
- Enseguida, se realiza una extracción del ruido y los errores surgidos de la etapa pre procesamiento.
- Luego, se utiliza el algoritmo de OCR para convertir la imagen a texto.
- Posteriormente, se analiza el texto obtenido del algoritmo de OCR y se valida si el resultado es correcto o erróneo.
- Finalmente, mediante una etapa de visualización de resultados se alerta al operador de la presencia de una falla o problemas en el sistema de etiquetado.

La implementación y desarrollo de los algoritmos propuestos en este sistema se realizó utilizando el siguiente software de fuente abierto:

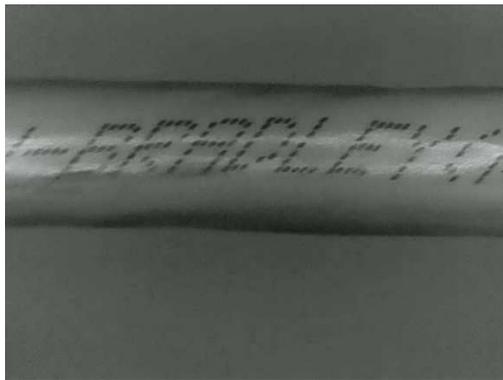
- Lenguaje de programación Python [78].
- Librerías de visión por computadora OpenCV (en inglés, Open Source Computer Vision Library) [79].
- Ambiente de programación PyCharm [80].
- El banco de trabajo para aprendizaje automático WEKA [81], [82].
- La colección de algoritmos para el procesamiento de imágenes scikit-image [83], llamada también Skimage.

## **4.2 SISTEMA PROTOTIPO PARA LA ADQUISICIÓN DE IMÁGENES**

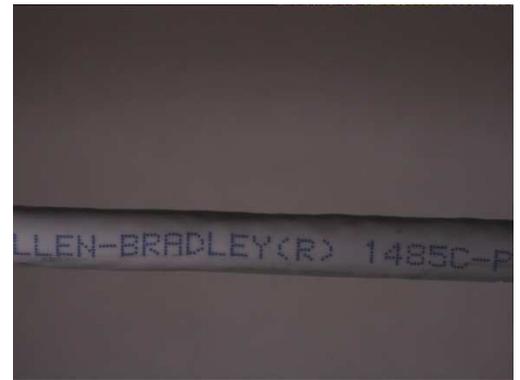
A la par con el desarrollo de esta tesis se desarrolló un sistema prototipo de adquisición de imágenes y de iluminación que permite realizar una captura lo más cercana a las

condiciones reales que se presentan en planta. Para dicho desarrollo se tomaron en cuenta los parámetros y consideraciones detalladas a continuación:

- Los procesos que se utilizan para fabricar conductores eléctricos, por lo general, trabajan a alta velocidad, dependiendo del calibre del conductor eléctrico; por lo cual, como se explicó en el capítulo 2.4, es necesario considerar la deformación por rolling shutter. Para solucionar la deformación por rolling shutter se seleccionó una cámara con obturador global; este tipo de cámara permite capturar todos los píxeles de la imagen al mismo tiempo, generando una imagen sin la presencia de la deformación óptica. La Figura 28 presenta imágenes capturadas de un objeto en movimiento con una cámara con obturador de barrido y con una cámara con obturador global.



(a)



(b)

Figura 28. (a) Imagen capturada con una cámara con obturador por barrido;  
(b) Imagen capturada con una cámara con obturador global

Fuente: elaboración propia

En la Figura 28 puede apreciarse visualmente que la imagen capturada con la cámara con obturador global no presenta el problema de deformación óptica mostrado en la imagen capturada con la cámara con obturador de barrido.

- La iluminación de un sistema de adquisición de imágenes juega un papel importante dentro del mismo, ya que permite mejorar las condiciones de la imagen a capturar. Dentro de la iluminación del sistema fue necesario considerar

la forma física del objeto a analizar. En esta tesis, al tener forma cilíndrica el objeto de estudio, su superficie no es regular, por lo cual el sistema de iluminación debe evitar generar sombras y brillos a lo largo del cable.

En este documento no se darán detalles del prototipo de captura de imágenes digitales, pues se considera material confidencial, debido a que será sometido a patente. Sin embargo, las características de las imágenes adquiridas se presentan en el capítulo 5.1 las cuales formaron una base de datos para la evaluación del desempeño del algoritmo.

### 4.3 ALGORITMO DE PRE PROCESAMIENTO

La etapa de pre procesamiento permite mejorar las características de la imagen que se ingresa al algoritmo de OCR y su finalidad es disminuir la tasa de error producida en la detección de texto por defectos en la imagen o ruido. Para la solución de esta etapa se desarrollaron y compararon tres algoritmos con base en distintos métodos. El primer método se basa en operaciones morfológicas, denominado TBH Engine, (en español, Motor TBH), el segundo se basa en operaciones de clusterizado, denominado KMN Engine (en español, motor KMN) y el tercero se basa en reconstrucción morfológica, denominado MrRecon Engine (en español, motor MrRecon). Los pasos generales para cada uno de los algoritmos se detallan a continuación.

#### 4.3.1 Método basado en operaciones morfológicas, TBH Engine

##### Paso 1. Cambio de espacio de color LAB

La imagen de entrada es del tipo RGB por lo que se realiza el cambio al espacio CIE L\*a\*b\*. Este espacio está definido por el conjunto de Ecuaciones 4 [84].

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.412453 & 0.357580 & 0.180423 \\ 0.212671 & 0.715160 & 0.072169 \\ 0.019334 & 0.119193 & 0.950227 \end{bmatrix} \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

$$X \leftarrow \frac{X}{X_n} \text{ donde } X_n = 0.950456$$

$$Z \leftarrow \frac{Z}{Z_n} \text{ donde } Z_n = 1.088754$$

$$L \leftarrow \begin{cases} 116 * Y^{\frac{1}{3}} - 16 & Y > 0.008856 \\ 903.3 * Y & Y \leq 0.008856 \end{cases} \quad (4)$$

$$a \leftarrow 500(f(X) - f(Y)) + \mathit{delta}$$

$$b \leftarrow 200(f(X) - f(Z)) + \mathit{delta}$$

$$\text{Donde } f(t) = \begin{cases} t^{\frac{1}{3}} & \text{para } t > 0.008856 \\ 7.78t + \frac{16}{116} & \text{para } t \leq 0.008856 \end{cases}$$

$$\text{y } \mathit{delta} \leftarrow \begin{cases} 128 & \text{img de 8 bits} \\ 0 & \text{img punto flotante} \end{cases}$$

$$\text{Imágenes de 8 bits: } L \leftarrow L * \frac{255}{100}, a \leftarrow a + 128, b \leftarrow b + 128$$

En el método propuesto se utiliza solamente el canal  $L^*$  correspondiente al canal de luminosidad, permitiendo reducir el número de canales a procesar.

Cabe mencionar que se analizaron otros espacios de color como escala de grises y HSV. Sin embargo, experimentalmente se determinó que la imagen representada en  $L^*$  proporciona una mejor variación entre los píxeles claros y oscuros con respecto a los canales de los otros espacios de color. En esta tesis la imagen original en RGB es denotada como **IMG** y la imagen correspondiente a este paso del método se denota como **IMG<sub>L</sub>**. En la Sección 5.3.1 se presenta una discusión sobre esta decisión.

## **Paso 2.** Extracción del texto

Para la extracción del texto contenido en **IMG<sub>L0</sub>** se realizó una operación morfológica. Dependiendo del fondo de la imagen se generaron dos casos:

- Cuando el fondo de la imagen es de color oscuro, la operación a realizar corresponde a la de Black-Hat (en español, sombrero de copa negro). Esta operación se define en la Ecuación 5:

$$BH(f) = f - f \circ B \quad (5)$$

Donde  $\bullet$  representa la operación de apertura.

- Cuando el fondo de la imagen es de color claro, la operación a realizar corresponde a la de Top-Hat (en español, sombrero de copa blanco). Esta operación se define en la Ecuación 6:

$$TH(f) = f - f \bullet B \quad (6)$$

Donde  $\bullet$  representa la operación de cerradura.

En ambas ecuaciones,  $f$  corresponde a una imagen, en este caso es  $IMG_L$  y al tratarse de una imagen en el canal L\* contiene valores entre 0 y 100, los cuales son escalados por la misma librería de OpenCV [84] al rango de valores entre 0 y 255 correspondientes a una imagen de 8 bits;  $B$  es un elemento estructural definido por la Ecuación 7:

$$B = \begin{bmatrix} 0 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 0 \end{bmatrix}_{M \times M} ; M = 81 \quad (7)$$

$$\text{donde el renglón } B \begin{bmatrix} M \\ 2 \end{bmatrix} = [1 \quad \dots \quad 1]$$

Para este caso de estudio,  $B$  se definió como una matriz de ceros de tamaño  $M \times M$ , con el renglón  $B \begin{bmatrix} M \\ 2 \end{bmatrix}$  compuesto por un renglón de unos. El tamaño de  $M$  y la forma de  $B$  se discuten en la Sección 5.3.2.

El resultado de la operación de  $BH$  o  $TH$  esta denotado como  $IMG_{MO}$ . La Figura 29 presenta el resultado de aplicar la operación de  $BH$  o  $TH$  a dos imágenes de ejemplo.



Figura 29. Resultado de la operación morfológica y la imagen original. (a) imagen RGB antes de aplicar Top-Hat; (b) imagen RGB antes de aplicar Black Hat; (c) resultado de Top-Hat; (d) resultado de Black-Hat

Fuente: elaboración propia

**Paso 3.** Suavizado de la imagen utilizando un filtro de mediana.

El suavizado de la imagen  $IMG_{M0}$  se realizó utilizando un filtro de mediana [85], permitiendo reducir pequeños puntos de ruido generados por el paso anterior. La imagen generada en este punto esta denotada como  $IMG_{MF}$

**Paso 4.** Binarización de la imagen utilizando el método de umbralización automático de Otsu.

En este paso se utiliza el método de Otsu [86] debido a que al ser un método de umbralizado automático permite separar la información de  $IMG_{MF}$  en dos clases. Estas clases se definen en la Ecuación 8:

$$IMG_{bin}(x, y) = \begin{cases} 255 & IMG_{MF}(x, y) > trh \\ 0 & IMG_{MF}(x, y) \leq trh \end{cases} \quad (8)$$

Donde  $x, y$  corresponden a las coordenadas de la imagen,  $trh$  corresponde al valor limite obtenido del método de Otsu. Por lo tanto,  $IMG_{bin}$  corresponde a una imagen binaria en donde los valores mayores a  $trh$  se convierten a blanco (255) y los valores menores se convierten a negro (0). La Figura 30 muestra el resultado de este paso.

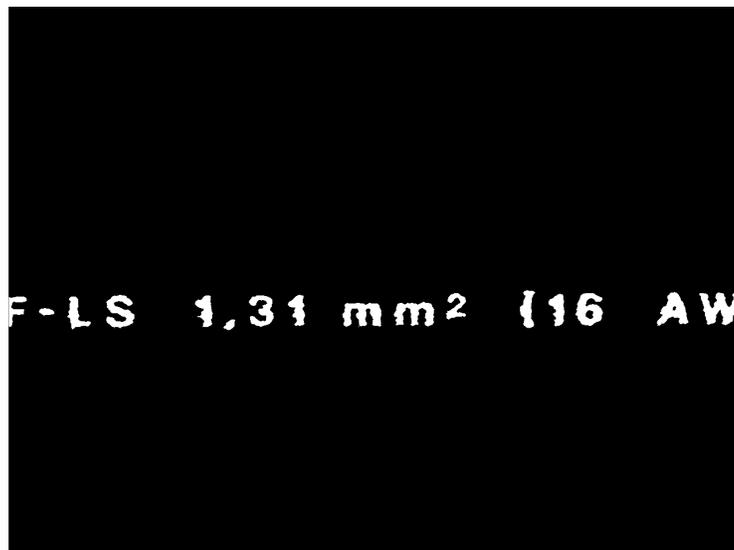


Figura 30. Imagen binarizada por el método de Otsu

Fuente: elaboración propia

**Paso 5.** Operación lógica AND.

La imagen obtenida en el paso 4  $IMG_{bin}$  se utilizó como máscara para realizar una operación de conjunción lógica ( $\wedge$  o *AND*), elemento por elemento, definida en la Ecuación 9:

$$IMG_{bw} = IMG_V \wedge IMG_V \quad \text{donde } IMG_{bin}(x, y) = 255 \quad (9)$$

La operación de conjunción lógica  $\wedge$  junto con la restricción de  $IMG_{bin}$ , permiten solamente extraer las regiones donde la imagen binarizada equivale a 255 (funcionando como una máscara). La Figura 31 muestra el resultado de la operación de conjunción.



Figura 31. Conjunción lógica AND utilizando una máscara  
Fuente: elaboración propia

**Paso 6.** Reducción del borde en la imagen con texto segmentado.

En  $IMG_{bw}$  aquellos pixeles cuyo valor es diferente de 0 corresponden al texto a extraer en la imagen  $IMG_V$ . Sin embargo, estos valores contienen pixeles pertenecientes al cable, por lo que se buscó reducir la cantidad de estos pixeles que a su vez son considerados ruido.

Para eliminar el ruido se calculó la media de los valores mayores a cero, definidos en la Ecuación 10:

$$\bar{x} = \frac{\sum IMG_{bw}[IMG_{bw}>0]}{N} \quad (10)$$

En donde  $N$  corresponde al número de pixeles en  $IMG_{bw}$  y  $\bar{x}$  corresponde al promedio de los pixeles mayores a cero en  $IMG_{bw}$ . Para poder reducir el borde de la imagen se definieron dos casos:

- Cuando el fondo de la imagen es claro se realizó una operación de umbralizado definida en la Ecuación 11:

$$mask_{brdr}(x, y) = \begin{cases} 0 & IMG_{bw}(x, y) > \bar{x} \\ 255 & IMG_{bw}(x, y) \leq \bar{x} \end{cases} \quad (11)$$

- Cuando el fondo de la imagen es oscuro se realizaron las siguientes operaciones:

$$brd(x, y) = \begin{cases} 0 & IMG_{bw}(x, y) > \bar{x} \\ IMG_{bw}(x, y) & IMG_{bw}(x, y) \leq \bar{x} \end{cases} \quad (12)$$

$$inv = \neg brd \quad (13)$$

$$mask_{brdr}(x, y) = \begin{cases} 255 & inv(x, y) > trh \\ 0 & inv(x, y) \leq trh \end{cases} \quad (14)$$

Donde  $\neg brd$  corresponde a la operación lógica de inversión,  $trh$  al igual que en la Ecuación 8, corresponde al umbral obtenido por el método de Otsu.

En la Figura 32 se presentan las máscaras obtenidas dependiendo del fondo.

XON TF-LS RAD ROHS 1,31 m

(a)

VIA XON TMW-2-LS / TMW-LS RAD 1,31 m

(b)

Figura 32. (a) máscara obtenida por la Ecuación 11: (b) máscara obtenida por las Ecuaciones 12, 13 y 14

Fuente: elaboración propia

Una vez obtenida la máscara para la reducción del borde se realizó una operación de conjunción lógica AND definida en la Ecuación 15:

$$IMG_{bdl} = IMG_{bw} \wedge mask_{brdr} \quad (15)$$

En la Figura 33 se presenta el resultado de la Ecuación 15, en donde los pixeles correspondientes al cable han sido reducidos.



(a)



(b)

Figura 33 – Reducción del borde en la imagen con texto segmentado.

(a) imagen  $IMG_{bw}$ ; (b) borde mejorado en imagen anterior

Fuente: elaboración propia

**Paso 7.** Suavizado de la imagen utilizando un filtro de mediana.

El suavizado de la imagen  $IMG_{bdl}$  se realizó utilizando un filtro de mediana [85], [87]. Esto permitió reducir pequeñas zonas de ruido en los cuales su valor es igual al valor de los límites. La imagen generada en este paso se denota como  $IMG_{Mseg}$

La imagen resultante del paso 7  $IMG_{Mseg}$  es la imagen resultante del TBH Engine.

#### 4.3.2 Método basado en clusterización, KMN Engine

**Paso 1.** Cambio de espacio de color CIE  $L^*a^*b^*$ .

La imagen de entrada se encuentra en el espacio de color RGB por lo que se realizó el cambio de espacio de color a CIE  $L^*a^*b^*$ , utilizando el conjunto de Ecuaciones 4 presentadas en la Sección 4.3.1. En este método propuesto se utiliza solamente el canal  $L^*$  correspondiente al

canal de luminosidad, permitiendo reducir el número de canales a procesar. En esta tesis la imagen original en RGB es denotada como **IMG** y la imagen correspondiente a este paso se denota como **IMG<sub>L</sub>**.

**Paso 2.** Clusterización por K-medias [46].

Para la extracción del texto contenido en **IMG<sub>L</sub>**, se utilizó el método de clusterización de k-medias [46] incluido en la librería de OpenCV [88]. Se definieron cuatro clusters, a los cuales se le asignaron el valor del centroide de los píxeles contenidos en los mismos, permitiendo reducir así el número de valores de intensidad en la imagen a cuatro valores de intensidad. En la Figura 34 se presenta una imagen que muestra los grupos obtenidos coloreados con un color distinto.

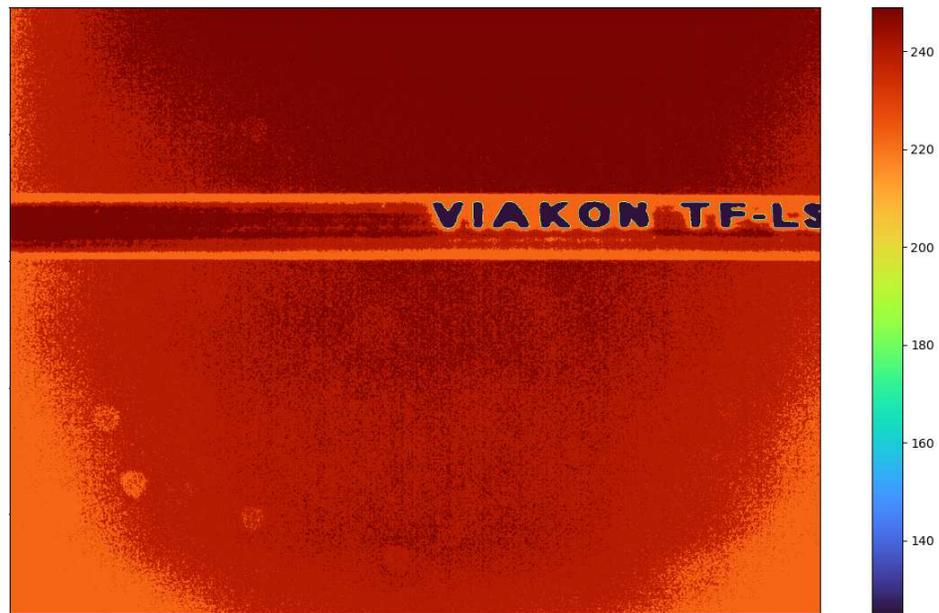


Figura 34. Resultados del clusterizado, cada grupo está representado con un color distinto

Fuente: elaboración propia

En la Figura 34 del lado derecho se presenta la barra de color que define el valor del píxel y su color; los píxeles más oscuros corresponden al texto en la imagen; el cable y el fondo de la imagen, al estar entre los valores superiores de la barra de color, corresponden a colores cercanos al

blanco. La imagen resultante de este paso se denota como  $IMG_{KMN}$ . En el capítulo 5.3.3 se discute la elección del número de clusters utilizados.

**Paso 3.** Obtención de los valores únicos.

De la imagen obtenida en el paso 2, se obtienen los valores únicos en la imagen; el resultado es almacenado en una lista denominada **Values**, en donde cada uno de los valores contenidos en ella corresponde a cada uno de los grupos creados en el paso anterior.

Por ejemplo:

$$Values = [125,222,239,249]$$

**Paso 4.** Umbralización.

El siguiente paso consiste en realizar un umbralizado a la imagen  $IMG_{KMN}$ , utilizando los valores obtenidos en **Values**. Para determinar el valor a utilizar se generaron dos casos:

- Cuando el fondo de la imagen es claro se utilizó el primer valor de **Values**, que corresponde al grupo de colores más oscuros, en este caso, el texto a segmentar. Para este caso, el umbralizado se define por la Ecuación 16.

$$text_{val} = Values [0]$$

$$mask_{kmn}(x, y) = \begin{cases} 0 & IMG_{KMN}(x, y) \neq text_{val} \\ 255 & IMG_{KMN}(x, y) = text_{val} \end{cases} \quad (16)$$

- Cuando el fondo de la imagen es oscuro se utilizó el ultimo valor de **Values**, que corresponde al grupo de colores más claros, en este caso, el texto a segmentar. Para este caso, el umbralizado se define por la Ecuación 17.

$$text_{val} = Values[3]$$

$$mask_{kmn}(x, y) = \begin{cases} 0 & IMG_{KMN}(x, y) \neq text_{val} \\ 255 & IMG_{KMN}(x, y) = text_{val} \end{cases} \quad (17)$$

**Values[0]** representa el primer valor de la lista, **Values[3]** representa el último valor de la lista  $(x, y)$  que contiene los valores de intensidad de los píxeles en  $IMG_{KMN}$ . En la Figura 35 se presentan las máscaras obtenidas dependiendo del color de fondo en cada caso.



(a)

(b)

Figura 35. (a) Máscara obtenida por la Ecuación 16 cuando el fondo es claro; (b) máscara obtenida por la Ecuación 17 cuando el fondo es oscuro

Fuente: elaboración propia

**Paso 5.** Operación lógica AND con máscara.

La máscara obtenida en el paso 4, definida como  $mask_{kmn}$ , se utilizó para realizar una operación de conjunción lógica ( $\wedge$  o **AND**), elemento por elemento, definida por:

$$IMG_{Kmask} = IMG_{KMN} \wedge IMG_{KMN} \quad \text{donde } mask_{kmn}(x, y) = 255$$

La operación de conjunción lógica **AND** junto con la restricción de la máscara permiten extraer solamente las regiones de la imagen  $IMG_{KMN}$  que coinciden con la máscara.

**Paso 6.** Suavizado de la imagen utilizando un filtro de mediana.

Finalmente, para suavizar la imagen  $IMG_{Kmask}$  se utilizó un filtro de mediana [85], [87]. Esto permitió reducir pequeñas zonas de ruido generadas en los pasos anteriores. La imagen resultante en este paso se denota como  $IMG_{KSeg}$

La imagen resultante del paso 6  $IMG_{KSeg}$  es la imagen resultante del KMN Engine. La Figura 36 presenta un ejemplo del resultado de dicho método.

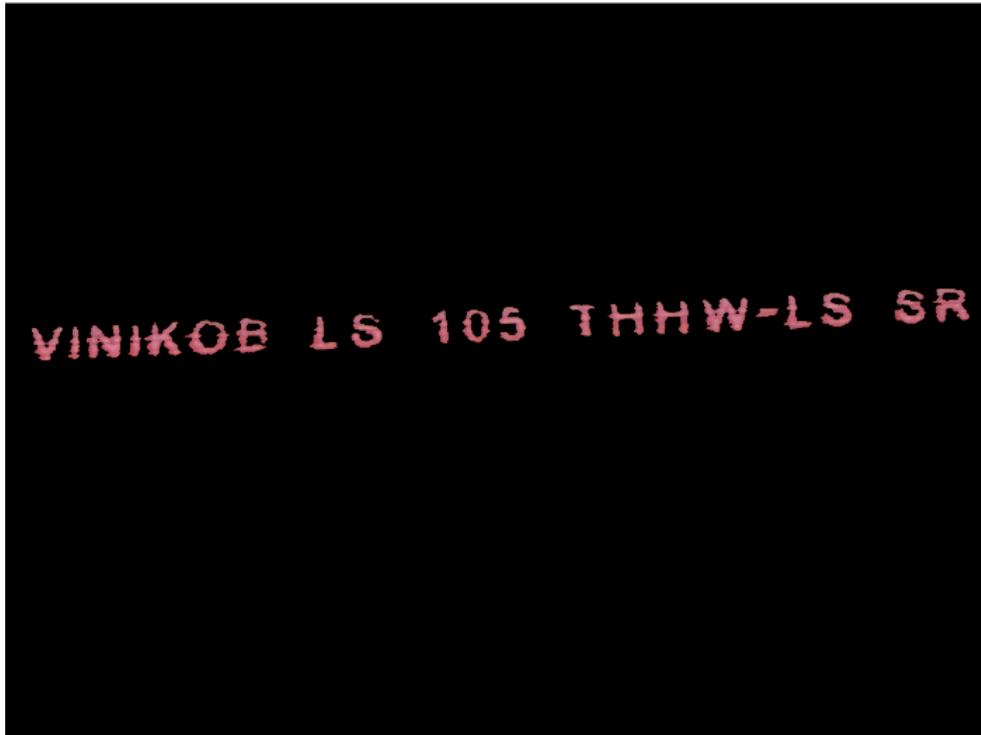


Figura 36. Resultado del algoritmo de clusterización

Fuente: elaboración propia

#### 4.3.3 Método basado en reconstrucción morfológica, MrRecon Engine

**Paso 1.** Cambio de espacio de color CIE  $L^*a^*b^*$ .

La imagen de entrada se encuentra en el espacio de color RGB por lo que se realizó el cambio de espacio de color a CIE  $L^*a^*b^*$ , utilizando el conjunto de Ecuaciones 4 presentadas en la Sección 4.3.1. En este método propuesto se utiliza solamente el canal  $L^*$  correspondiente al canal de luminosidad, permitiendo reducir el número de canales a

procesar y trabajar en un canal con mayor distancia en el valor de los píxeles oscuros y claros. La imagen resultante de este paso se denota como  $IMG_L$ .

**Paso 2.** Inversión de valores.

Se definieron dos casos para procesar la imagen, los cuales dependen de la intensidad del fondo de la imagen  $IMG_L$ . Cuando el fondo es blanco se realiza una operación de inversión bit a bit de la imagen de entrada, definida por la Ecuación 18.

$$IMG_{Inv} = \neg IMG_L \quad (18)$$

En donde  $\neg$  corresponde a la operación lógica de inversión.

Cuando el fondo es oscuro la imagen se mantiene igual, definido en la Ecuación 19.

$$IMG_{inv} = IMG_L \quad (19)$$

**Paso 3.** Conversión de imagen a flotante.

Utilizando las funciones incluidas en la biblioteca skimage, se convierte la imagen de 8 bits  $IMG_{inv}$  a punto flotante, lo que significa que la imagen ahora contiene un rango de valores entre [0.0 y 1.0] en formato flotante de 64 bits, permitiendo así tener un mayor rango de valores que los 8 bits iniciales. La imagen en este paso esta denotada como  $IMG_{float}$ .

**Paso 4.** Suavizado de la imagen utilizando un filtro Gaussiano.

La imagen  $IMG_{float}$  es suavizada utilizando un filtro Gaussiano [88], para poder atenuar los bordes de la imagen y el ruido presentado en la imagen. La imagen resultante de este paso esta denotada como  $IMG_{fsg}$

**Paso 5.** Creación de la semilla y máscara de reconstrucción morfológica.

En la reconstrucción morfológica por dilatación[89] los valores de alta intensidad remplazarán a los valores vecinos de baja intensidad. Sin

embargo, el operador de dilatación utiliza un elemento estructurante para determinar hasta qué punto se puede extender un valor en la imagen. En este caso, al realizar una reconstrucción se utilizan dos imágenes: una imagen semilla  $IMG_{seed}$ , que especifica los valores que se extienden y una imagen máscara  $IMG_{mask}$ , la cual especifica el valor máximo permitido en cada píxel. El utilizar la imagen máscara como elemento estructurante, limita la propagación de valores de alta intensidad.

Por lo tanto, las imágenes de semilla y máscara se definieron por las Ecuaciones 20, 21 y 22 de la siguiente manera:

$$IMG_{mask} = IMG_{fsg} \quad (20)$$

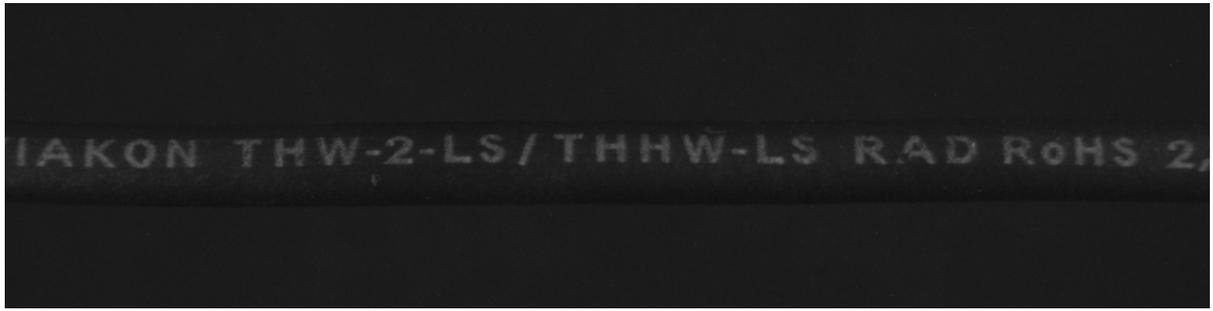
$$IMG_{seed} = IMG_{fsg} \quad (21)$$

$$\text{pero } IMG_{seed}[fe: le_x, fe: le_y] = \min(IMG_{fsg}) \quad (22)$$

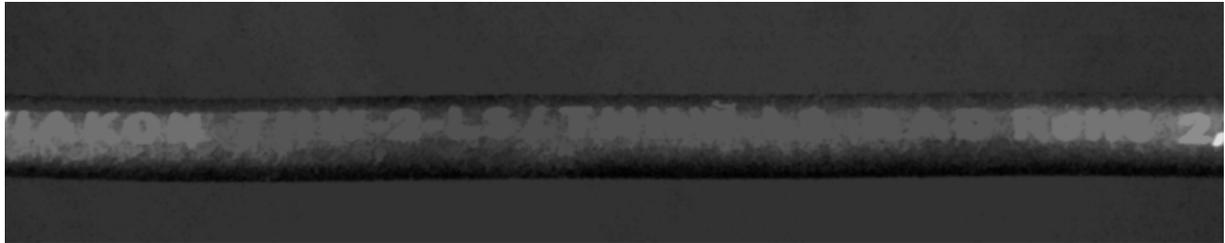
En donde  $fe$  corresponde al índice del segundo elemento de la imagen y  $le$  corresponde al índice del penúltimo elemento de la imagen;  $le_x$  y  $le_y$  pueden tener distintos valores dependiendo del ancho y alto de la imagen. La imagen máscara  $IMG_{mask}$  se define como una copia de la imagen de entrada; la imagen semilla  $IMG_{seed}$  se inicializa con la intensidad mínima de la imagen de entrada (Ecuación 22); por lo tanto, el borde de la semilla mantiene el mismo valor de la imagen  $IMG_{fsg}$ .

#### **Paso 6.** Reconstrucción de la imagen.

Teniendo las imágenes de  $IMG_{seed}$  e  $IMG_{mask}$ , el siguiente paso consiste en reconstruir la imagen utilizando la función de la biblioteca de skimage. Esta función utiliza las imágenes del paso 5 y definiendo como método de reconstrucción el de la dilatación se obtiene la imagen denotada como  $IMG_{recon}$ . La Figura 37 presenta un ejemplo de la imagen resultante y la comparación con la imagen original  $IMG_{fsg}$ .



(a)



(b)

Figura 37. (a) Imagen original; (b) imagen resultante de la reconstrucción morfológica por dilatación

Fuente: elaboración propia

En la Figura 37 (b) el texto al contener un valor mayor al inicializado en la semilla no es reconstruido, a excepción del texto que se encuentra ubicado en el borde de la imagen, el cual coincide con los valores iniciales de la semilla.

#### **Paso 7.** Resta de imágenes

La imagen obtenida en el paso 6  $IMG_{recon}$ , carece de los píxeles que equivalen al texto. Esto puede ser utilizado para extraer de la imagen original el texto utilizando una resta, la cual se definió en la Ecuación 23 como:

$$Recon_{img64} = IMG_{fsg} - IMG_{recon} \quad (23)$$

La Figura 38 muestra un ejemplo del resultado de la operación de resta de la Ecuación 23 utilizada para reconstruir la imagen.



Figura 38. Resultado de la resta de las imágenes en la Ecuación 23, la cual permite segmentar el texto de la imagen original

Fuente: elaboración propia

En la Figura 38 se puede observar que el resultado de la Ecuación 23 extrae el texto de la imagen, ya que estos valores no fueron reconstruidos en el paso 6 por tener valores mayores a la semilla.

**Paso 8.** Conversión de la imagen a 8-bits.

La imagen del paso anterior mantiene el formato de 64 bits, por lo tanto, para reducir el espacio en memoria que utiliza y debido a que la mayoría de las funciones de la biblioteca de OpenCV trabajan con imágenes de 8 bits, la imagen **Recon<sub>img64</sub>** es convertida de regreso a una imagen de 8 bits. La imagen resultante este paso es denotado como **IMG<sub>GRSeg</sub>**.

**Paso 9.** Umbralización y operación lógica AND

En este paso se utiliza el método de Otsu [86] debido a que al ser un método de umbralizado automático permite separar la información de **IMG<sub>GRSeg</sub>** en dos clases, utilizando la Ecuación 8 presentada en la Sección 4.3.2 . La imagen resultante de la umbralización es denotada como **mask<sub>BinSeg</sub>**. Esta imagen, se utilizó para realizar una operación de conjunción lógica ( $\wedge$  o **AND**), elemento por elemento, definida por:

$$IMG_{MrRecon} = IMG_L \wedge IMG_L \text{ donde } mask_{BinSeg}(x, y) = 255$$

La operación de conjunción lógica **AND** junto con la restricción de la máscara permiten extraer solamente las regiones de la imagen **IMG<sub>L</sub>** que coinciden con la máscara.

La imagen resultante del paso 9 **IMG<sub>MrRecon</sub>** es la imagen resultante del MrRecon Engine.

#### 4.3.4 Fusión de métodos

Los métodos propuestos TBH, KMN y MrRecon utilizan un principio de funcionamiento diferente para la segmentación del texto. Por lo tanto, para buscar crear un sistema, el cual sea robusto ante los diferentes colores de los cables y del texto impreso y que a la vez no presente falsos negativos, se decidió crear una fusión de los métodos. La forma en la que estos métodos fueron fusionados se desarrolla después de que las imágenes resultantes de cada uno de los métodos han sido procesadas por el algoritmo VALET (descrito en la sección 4.6.3). El resultado de VALET es una lista con los valores que han sido clasificados correctamente; por lo tanto, es posible fusionar las tres listas generadas de cada método para poder validar algunos caracteres que no fueron reconocidos con un método, pero con otro sí. La fusión de métodos simula un sistema de votación, en donde además de tener la certeza de que los caracteres detectados son válidos, también se permiten conocer aquellos caracteres en los cuales se están presentando problemas.

#### 4.4 ALGORITMO PARA DETECCIÓN DE RUIDO Y DEFECTOS EN EL TEXTO (S.A.F.E.)

Dentro de la información que se extrae en la etapa de pre procesado utilizando los tres algoritmos se encuentra también información que se puede considerar ruido o errores de impresión, ya que estos llegan a tener el mismo tono o color del texto que se extrae de la imagen. Por este motivo, se desarrolló un algoritmo de extracción de errores y ruido en la impresión denominado **S.A.F.E.** (en inglés, noise and fault detector). La manera en la que se clasificó si una serie de píxeles que forman un contorno son ruido o errores de impresión está definida por las características del contorno mismo.

A partir de las regiones del texto segmentado se extrajeron distintas características para describirlas. Las características que se utilizaron para clasificar los contornos son:

- Hull (del inglés, cascara) [90] denotado como ***Hu***
- Área del contorno [90] denotado como ***A***
- Área de un rectángulo conteniendo al contorno [90] denotado como ***Ra***
- Perímetro del contorno [90] denotado como ***P***
- Relación de aspecto [91] denotado como ***RA***

- Compacidad definida como:  $CMP = \frac{p^2}{Ra}$  (24)

- Extensión definida como  $ET = \frac{A}{Ra}$  (25)

- Área del Hull [91] denotado como  $HA$

También se definieron tres clases de contornos:

- Clase 1: Contornos correspondientes a texto.
- Clase 2: Contornos correspondientes a símbolos gráficos en el texto, por ejemplo: coma (,), guión (-) y super índices (<sup>2</sup>), etc.
- Clase 3: Contornos correspondientes a ruido o errores de impresión.

La Figura 39 presenta un ejemplo de las tres clases de contornos.



Figura 39. Representación de las tres clases de contornos en una sola imagen

Fuente: elaboración propia

En la Figura 39 el contorno en color verde correspondiente a la clase 1, los contornos correspondientes a la clase 2 se muestran en color azul y los contornos correspondientes a la clase 3 se muestran de color amarillo. De cada una de estas clases se obtuvieron las características anteriormente mencionadas. Considerando estas características se desarrolló un dataset (del inglés, base de datos) formado por 2023 muestras. Se utilizó el software WEKA [81] para examinar los datos con los algoritmos de clasificación incluidos en este software y para generar un modelo de clasificación que permita discriminar entre clases de contorno.

El algoritmo utilizado para obtener el modelo de clasificación es el JRIP[82], el cual implementa un aprendizaje de reglas proposicionales basado en el algoritmo RIPPER [92]. El algoritmo JRIP generó un conjunto de reglas sencillas. La discusión sobre la elección del algoritmo JRIP se presenta en la Sección 5.3.4

Una vez obtenidas las reglas, para poder clasificar correctamente los contornos se plantean los pasos del algoritmo S.A.F.E.

**Paso 1:** Obtención de contornos.

Utilizando algoritmos de la librería de OpenCV, se obtienen los contornos de una imagen binaria  $IMG_{seg}$ , la cual corresponde a la imagen obtenida de los algoritmos anteriores Sección 4.3. Los contornos obtenidos en este paso se denotan como **Cnts**.

**Paso 2:** Obtención de las características de los contornos.

Para calcular las características de los contornos mencionadas a principio de esta sección se utilizaron las funciones implementadas en la librería de OpenCV[90], [91].

**Paso 3:** Clasificación de los contornos.

Los contornos se clasifican en tres clases distintas utilizando la lógica descrita en la Figura 40.

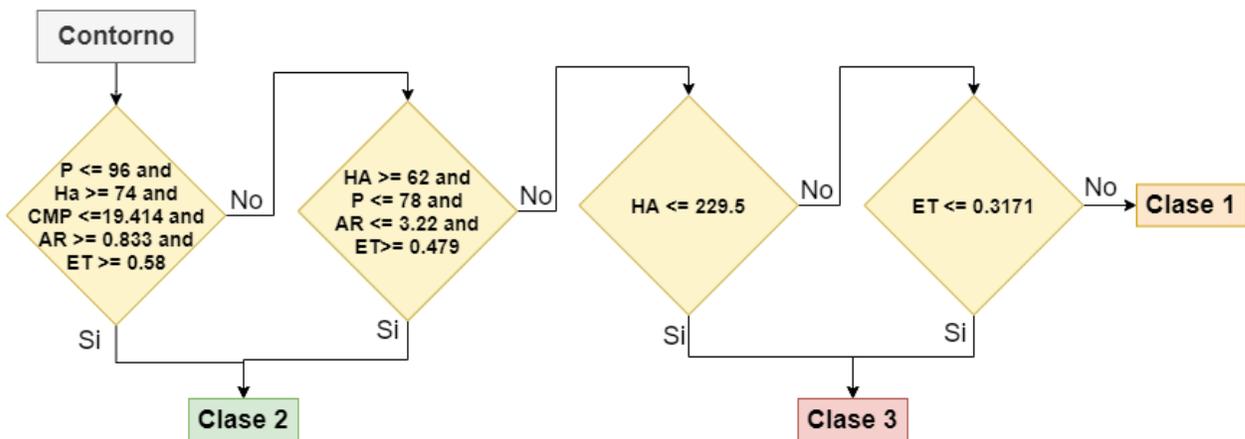


Figura 40. Reglas para clasificar los contornos

Fuente: elaboración propia

La Figura 40 describe las reglas utilizadas para clasificar cada uno de los contornos; cuando el contorno clasificado es del tipo clase 3, se guarda en una lista denominada  $WRN_{box}$ ; los contornos del tipo clase 2 y clase 1 son almacenados en una lista denominada **CNTS**, en donde se almacenan las coordenadas de esos contornos.

La Figura 41 presenta un ejemplo del resultado del algoritmo S.A.F.E.



Figura 41. Representación gráfica del algoritmo de extracción de errores y ruido

Fuente: elaboración propia

En la Figura 41, los errores y el ruido que se pueda presentar en la imagen son marcados con un rectángulo en amarillo, mientras que los contornos correspondientes al texto son marcados con rectángulos de colores y enumerados de izquierda a derecha.

#### 4.5 ALGORITMO DE OCR

Los algoritmos de OCR permiten convertir el texto en una imagen a caracteres (códigos de 8 bits, como los representados en tablas ASCII). El motor o algoritmo OCR adoptado en esta tesis se puede consultar en la Sección 2.6.2.1 Tesseract-OCR.

La imagen umbralizada generada por el algoritmo de pre procesamiento Secciones 4.3.1, 4.3.2 y 4.3.3 se procesa en el motor de OCR, obteniendo de él los siguientes datos:

- El número de palabras.
- Las coordenadas de los vértices del rectángulo que contiene a la palabra. Para cada una de las palabras reconocidas, se obtienen cuatro coordenadas, correspondientes a una caja alrededor de la palabra. Estas coordenadas representan la posición  $(x, y)$  de la esquina superior izquierda, así como el ancho y alto de la misma caja; con estas coordenadas se realiza el cálculo para contener el número de palabras totales.
- La confianza o certeza de la predicción, con valor entre 0 y 100%.
- La predicción del texto de la imagen representado como cadena de caracteres.

En la Tabla 4 se presenta un ejemplo de los datos obtenidos por el motor de OCR en el sistema propuesto para una imagen dada.

Tabla 4. Datos obtenidos del motor de OCR en el sistema propuesto

Level	page_num	block_num	par_num	line_num	word_num	left	top	width	height	conf	text
1	1	0	0	0	0	0	0	319	240	-1	nan
2	1	1	0	0	0	31	110	288	10	-1	nan
3	1	1	1	0	0	31	110	288	10	-1	nan
4	1	1	1	1	0	31	110	288	10	-1	nan
5	1	1	1	1	1	31	111	60	9	72	VIAKON
5	1	1	1	1	2	101	110	80	9	68	THW-2-LS/

En la Tabla 4 es posible observar, a partir de la columna 5, la información del número de palabra, las coordenadas de los vértices del rectángulo que contiene cada palabra, la confianza de cada palabra y la predicción del texto; esta información es utilizada en la etapa de validación de texto, así como en la de visualización de los resultados. La última columna, la cual corresponde al texto, cuando la detección no ha obtenido ningún resultado aparece el símbolo nan (en inglés, not a number). En el capítulo 5 se discutirá la configuración del motor de OCR.

## 4.6 VALIDACIÓN DEL TEXTO

En [93] se presentan diferentes métricas o maneras de evaluar los resultados obtenidos por el OCR; de manera global, la evaluación se puede llevar a cabo a nivel palabra o a nivel carácter. Para evaluar el desempeño del sistema propuesto se adoptaron las métricas que a continuación se presentan y se desarrolló un algoritmo para la validación del texto.

### 4.6.1 Distancia de edición de Levenshtein

La distancia de edición de Levenshtein [94] permite medir la diferencia entre dos cadenas de texto, obteniendo el número mínimo de operaciones necesarias para convertir una cadena en otra. Las operaciones permitidas pueden ser: inserción, eliminación o sustitución de caracteres.

La distancia entre dos cadenas de texto  $a, b$ , de tamaño  $|a|$  y  $|b|$ , respectivamente, está dada por  $lev_{a,b}(|a|, |b|)$  en la Ecuación 26.

$$lev_{a,b}(i,j) = \begin{cases} \max(i,j) & \text{si } \min(i,j) = 0 \\ \min \left( \begin{array}{l} lev_{a,b}(i-1,j) + 1, \\ lev_{a,b}(i,j-1) + 1, \\ lev_{a,b}(i-1,j-1) + 1_{(a_i \neq b_j)} \end{array} \right) & \text{de otra manera} \end{cases} \quad (26)$$

Donde

$$1_{(a_i \neq b_j)} = \begin{cases} 0 & \text{cuando } a_i \neq b_j \\ 1 & \text{cuando } a_i = b_j \end{cases}$$

#### 4.6.2 Distancia de edición de Hamming

La distancia de edición de Hamming [94] permite obtener la diferencia entre dos cadenas de texto, obteniendo el número mínimo de sustituciones necesarias para convertir una cadena de texto en otra.

La distancia entre dos cadenas de texto  $a, b$ , de tamaño  $n$ , está dada por  $d_H(a, b)$ , la cuál es el número de posiciones en donde  $a, b$  difieren, como se muestra en la Ecuación 27.

$$d_H(a, b) = \sum_{i=0}^n \delta(a_i, b_i), \text{ en donde } \delta(a_i, b_i) = \begin{cases} 0 & \text{cuando } a_i \neq b_i \\ 1 & \text{cuando } a_i = b_i \end{cases} \quad (27)$$

#### 4.6.3 Algoritmo de validación del texto, VALET

Complementario a las métricas de distancia, se desarrolló un algoritmo de validación del texto denominado VALET (acrónimo de VALidación de ETiquetas). Este algoritmo está diseñado específicamente para el caso de estudio, ya que la etiqueta a detectar está formada por una secuencia de imágenes, en donde en cada imagen de la secuencia se encuentra una sección o segmento del texto. Por lo tanto, utilizando solamente las métricas de distancia de Levenshtein y Hamming se pierde la posibilidad de validar cuáles caracteres del texto son los que están teniendo problemas o errores.

El algoritmo VALET presenta las siguientes características:

- Robustez ante falsos positivos.
- Posibilidad de validar segmentos separados de la etiqueta.
- Validación binaria.

El algoritmo VALET se desarrolla alrededor de un acumulador  $GT_{Accum}$  el cual es un vector de ceros de tamaño  $M$  correspondiente a la cantidad de caracteres contenidos en la etiqueta del GT (en inglés, Ground Truth). Cada producto contiene una longitud distinta. Por ejemplo:

**GT =**

'K'	'O'	'B'	'R'	'E'	'X'	' '	'T'	'F'	'-'	'L'	'S'	' '	'I'	'.'
'3'	'1'	' '	'm'	'm'	'2'	' '	'('	'1'	'6'	' '	'A'	'W'	'G'	' )'
' '	'6'	'0'	'0'	' '	'V'	' '	'9'	'0'	'o'	'C'				

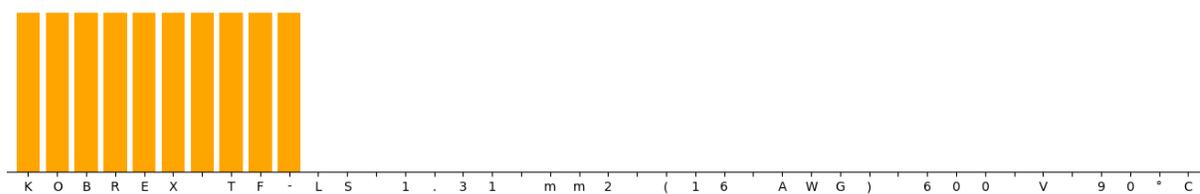
$\therefore GT_{accum} =$

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0				

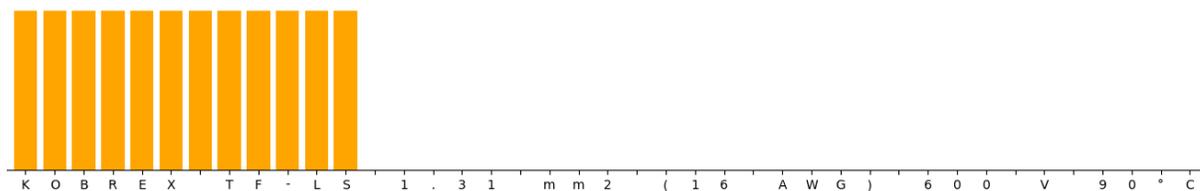
El resultado de la etapa de OCR (Sección 4.5) es ingresado al algoritmo VALET y cada carácter será considerado válido solamente si cumple con dos requisitos:

- Que el carácter sea el mismo carácter que el del GT.
- Que se encuentre en la misma posición conforme al vector  $GT_{Accum}$ .

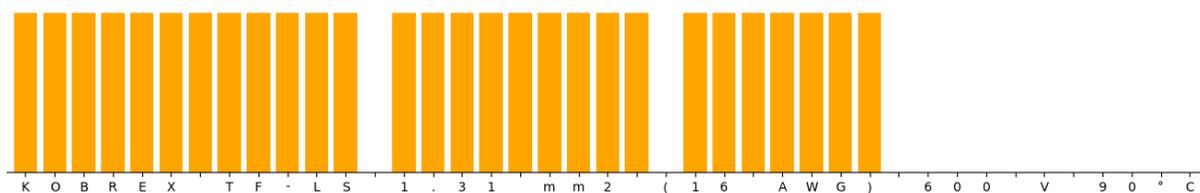
Cuando estos dos requisitos se cumplen, el acumulador  $GT_{Accum}$  cambia de cero a uno en la posición del carácter validado. Para facilitar la visualización del comportamiento del acumulador en la Figura 42 se presenta un ejemplo de la validación de una secuencia de imágenes.



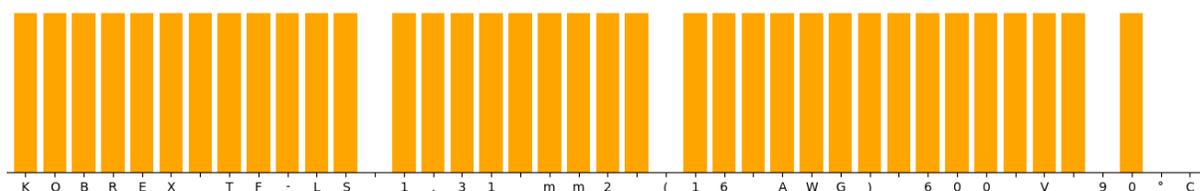
(a)



(b)



(c)



(d)

Figura 42. Secuencia de imágenes y su validación a través del algoritmo VALET. (a) primera imagen; (b) segunda imagen; (c) tercera imagen; (d) cuarta imagen

Fuente: elaboración propia

En la Figura 42 se presenta la validación del texto obtenido en la etapa de OCR utilizando el algoritmo VALET. En la primera imagen de la secuencia (a) el texto que es validado corresponde a los primeros 12 caracteres de la etiqueta completa. En la segunda imagen (b) solo se agregan dos caracteres más a la validación. En la tercera imagen (c) los caracteres que no contienen una barra de color corresponden a los caracteres que

no han sido validados, debido a un error de detección o a que presentan un problema de impresión. En la cuarta imagen (d) termina la secuencia y es posible observar aquellos caracteres no validados a lo largo de las imágenes de la secuencia. En caso de que estos mismos errores se repitan en secuencias posteriores, es posible alertar al usuario, sobre la presencia de errores o defectos en el etiquetado y especificar los caracteres en los cuales se están teniendo problemas. En la Sección 5.3.5 se discute la decisión sobre el número de secuencias necesarias para evitar generar falsos negativos.

#### 4.7 VISUALIZACIÓN DE RESULTADOS

En esta etapa se presenta de manera visual la información que se obtuvo con el algoritmo de OCR. En la Figura 43 se presenta un ejemplo de la visualización de los resultados obtenidos por el sistema de visión propuesto.



Figura 43. Resultado del motor OCR

Fuente: elaboración propia

En la Figura 43 se presenta la información que es extraída de los pasos anteriores, en orden de aparición de superior a inferior representan:

1. La certeza de cada una de las palabras
2. Las palabras obtenidas y agrupadas en un solo texto
3. Un rectángulo conteniendo a las palabras
4. Los valores calculados de las métricas de Levenshtein y Hamming, respectivamente

Para simplificar la visualización de resultados se creó la relación de colores mostrada en la Figura 43, la cual está compuesta por:

- Verde claro, texto correcto considerando espacios y palabras.
- Verde oscuro, texto correcto utilizando un comodín. La función del comodín se explica en la Sección 5.2.
- Naranja, texto correcto considerando solamente los caracteres.
- Rojo, texto incorrecto.

Este código de colores se aplica tanto en el rectángulo que contiene el texto detectado, como en el texto formado por la agrupación de las palabras obtenidas. La Figura 44 presenta ejemplos de imágenes que muestran el resultado de reconocimiento de texto con diferentes códigos de colores.



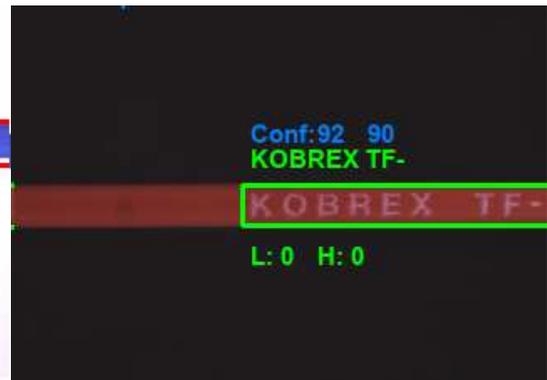
(a)



(b)



(c)



(d)

Figura 44. Código de colores. (a) texto correcto considerando solamente los caracteres; (b) texto correcto utilizando un comodín; (c) texto incorrecto; (d) texto correcto.

Fuente: elaboración propia

## CAPÍTULO 5. ANÁLISIS EXPERIMENTAL

En este capítulo se presenta el desarrollo y análisis de los experimentos llevados a cabo para la mejora y evaluación del método propuesto.

### 5.1 BASE DE DATOS DE IMÁGENES

Para el desarrollo de los diferentes experimentos realizados en esta tesis se desarrolló una base de datos formada por diferentes cables. La Tabla 5 presenta las características de cada uno de ellos. El total de imágenes para realizar pruebas es de 909 imágenes, formada por 8 distintos tipos de cable. La Figura 45 presenta una muestra de la base de los datos.

Tabla 5. Base de datos para desarrollo del análisis experimental

<b>Fabricante</b>	<b>Color del cable</b>	<b>Color del texto</b>	<b>Calibre</b>	<b>Número de imágenes</b>	<b>Imágenes con texto</b>	<b>Alias</b>
INDIANA	Rojo	Negro	12 AWG	74	60	Indiana Rojo
KOBREX	Azul	Blanco	14 AWG	71	54	Kobrex Azul
KOBREX	Rojo	Blanco	16 AWG	130	98	Kobrex Rojo
KOBREX	Rojo	Blanco	14 AWG	87	65	Kobrex Vinikob
VIAKON	Blanco	Negro	16 AWG	114	82	Viakon Blanco
VIAKON	Negro	Blanco	14 AWG	123	97	Viakon Negro
VIAKON	Azul	Negro	18 AWG	94	81	Viakon UL
VIAKON	Verde	Negro	14 AWG	216	170	Viakon Negro
			<b>Total de imágenes</b>	<b>909</b>	<b>707</b>	



(a)



(b)



(c)



(d)



(e)



(f)



Figura 45. Muestra de imágenes que conforman la base de datos. (a) Viakon Negro; (b) Viakon Blanco; (c) Kobrex Rojo; (d) Viakon UL; (e) Indiana Rojo; (f) Viakon Verde; (g) Kobrex Azul; (h) Kobrex Vinikob

Fuente: elaboración propia

Como se puede observar en la Figura 45, la base de datos está compuesta por cables que pueden ser similares en cuestión de color, el color del texto y el tipo de cable, así como el calibre. La base de datos contiene información variada que permiten evaluar el método propuesto en distintos casos. Cabe recordar que las imágenes incluyen características o problemas que se pueden presentar en planta, entre ellos se encuentran: suciedad en la impresora, desgaste o tallones en el cable, material de fabricación con mucho reflejo, corrimiento de tinta o gotas de tinta, y secciones del texto o partes de letras no impresas.

## 5.2 GROUND TRUTH

Para cada una de las imágenes de la base de datos se generó un archivo de ground truth (GT, en inglés base verdadera), el cual representa el texto real contenido en la imagen. La GT permite realizar una evaluación del desempeño del método propuesto contra el texto real contenido en cada imagen. El proceso de creación del ground truth se realizó de manera manual generando un archivo de texto, el cual contiene el texto contenido en la imagen.

Para mejorar los resultados obtenidos por el motor de OCR se decidió declarar un carácter comodín al principio o al final del ground truth, ya que, dentro del proceso de captura de la imagen, se puede cortar alguna de las letras iniciales o finales del texto. Realizar el análisis o la predicción del OCR sobre el texto con letras incompletas en gran porcentaje se considera una situación fuera de lo esperado para el motor OCR, afectando directamente al rendimiento del método. La Figura 46 muestra un ejemplo de una imagen la cual presenta la condición anteriormente descrita.

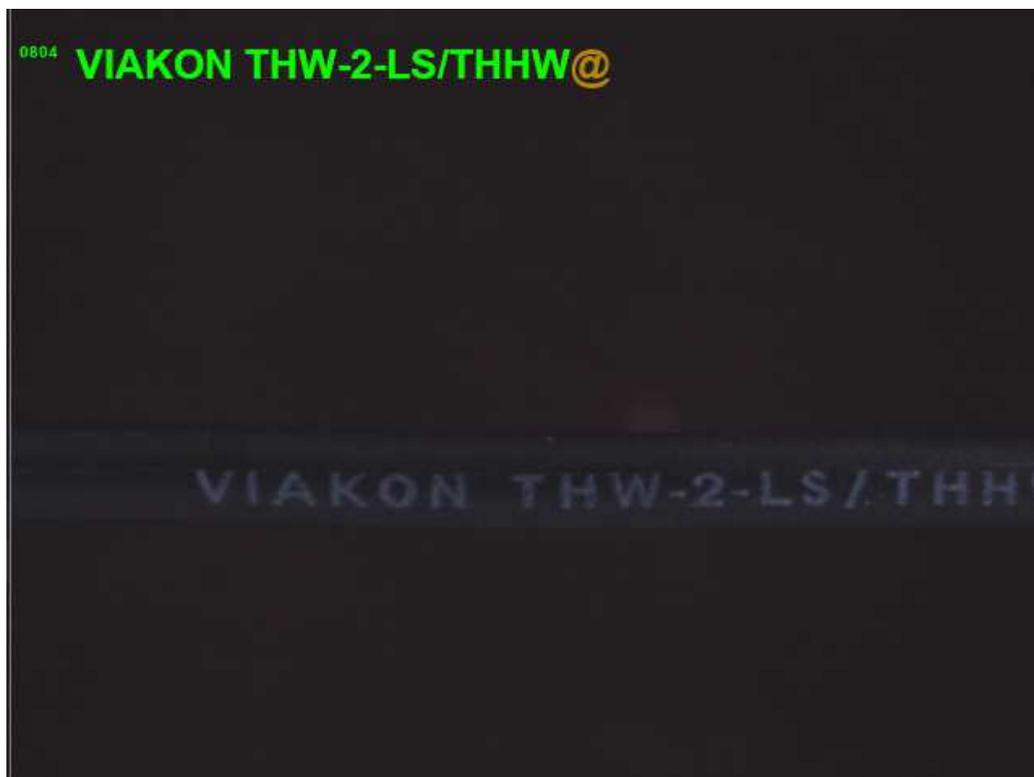


Figura 46. Imagen con comodín

Fuente: elaboración propia

En la Figura 46, el comodín está representado por el carácter @; este carácter puede estar colocado al principio, al final o en ambos extremos de la imagen, permitiéndole al método propuesto una alternativa para poder validar el texto del OCR como correcto. El número marcado en la imagen, con letras pequeñas a la izquierda de la etiqueta, corresponde al número de imagen dentro de la base de datos.

Se eligió un carácter comodín que no es utilizado en las etiquetas impresas en el texto. Las etiquetas pueden contener los 27 caracteres del alfabeto latino, así como símbolos

gráficos en el texto, por ejemplo: coma (,), guión (-) y super índices (<sup>2</sup>), etc. Por lo tanto, el símbolo de @ fue seleccionado debido a que el texto del cable a reconocer por el OCR no contiene este carácter.

### 5.3 PRUEBAS DE PRE PROCESAMIENTO

#### 5.3.1 Selección del espacio de color para el método propuesto TBH

A continuación, en la Tabla 6 se presenta una comparación entre tres espacios de colores distintos para determinar si existe una diferencia entre las características de cada uno utilizando el método de TBH.

Tabla 6. Comparación entre espacios de colores para método TBH

Conjunto	Promedio de aciertos Método TBH			Aciertos Totales por conjunto
	Escala de grises	Canal V de HSV	Canal L* de L*a*b*	
Indiana rojo	48.352	47.176	<b>49.411</b>	<b>50</b>
Viakon blanco	82.875	<b>85.125</b>	83.122	<b>91</b>
Kobrex azul	<b>61.444</b>	57.444	61.333	<b>83</b>
Kobrex rojo	<b>38.666</b>	37.444	38.555	<b>41</b>
Kobrex Vinikob	<b>68.363</b>	65.818	68.09	<b>83</b>
Viakon negro	<b>80.5</b>	77.071	79.5	<b>87</b>
Viakon UL	55.916	55.75	<b>56.5</b>	<b>68</b>
Viakon Verde	<b>83.363</b>	83.09	83.09	<b>87</b>
Sumatoria	519.479	508.918	<b>519.601</b>	<b>590</b>

La Tabla 6 se construyó con las 909 imágenes del dataset, validando el texto de la imagen con el algoritmo VALET y obteniendo el número de aciertos o caracteres correctos por imagen. En la Tabla 6 es posible observar que la diferencia entre el espacio de color de escalas de grises y el espacio de color CIE L\*a\*b\* es pequeña; considerando

que los otros dos métodos desarrollados KMN y MrRecon también trabajan con el espacio CIE L\*a\*b\*, este también será el espacio a trabajar para el método TBH.

### 5.3.2 Tamaño y forma del elemento estructural en el método propuesto TBH

En el método TBH el elemento estructural (abreviado como E. S.) define las características a extraer con la operación morfológica de Top/Black Hat. Para seleccionar el E. S. que permitan segmentar de manera correcta el texto de la imagen se plantearon experimentos donde se analizaron distintos tamaños y tipos de elemento estructural.

#### Experimento 1

El primer experimento consistió en comparar tres E. S. incluidos en OpenCV:

- Rect – E. S. rectangular, por ejemplo:

$$Rect_{2x2} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} \quad (28)$$

- Cross – E. S. en forma de cruz, por ejemplo:

$$Cross_{3x3} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \quad (29)$$

- Ellipse – E. S. el cual contiene una elipse inscrita en un rectángulo, por ejemplo:

$$Ellipse_{5x5} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \quad (30)$$

Estos E. S. se definieron en distintos tamaños; la Figura 47 presenta el resultado de aplicar estos E. S. en el método TBH.

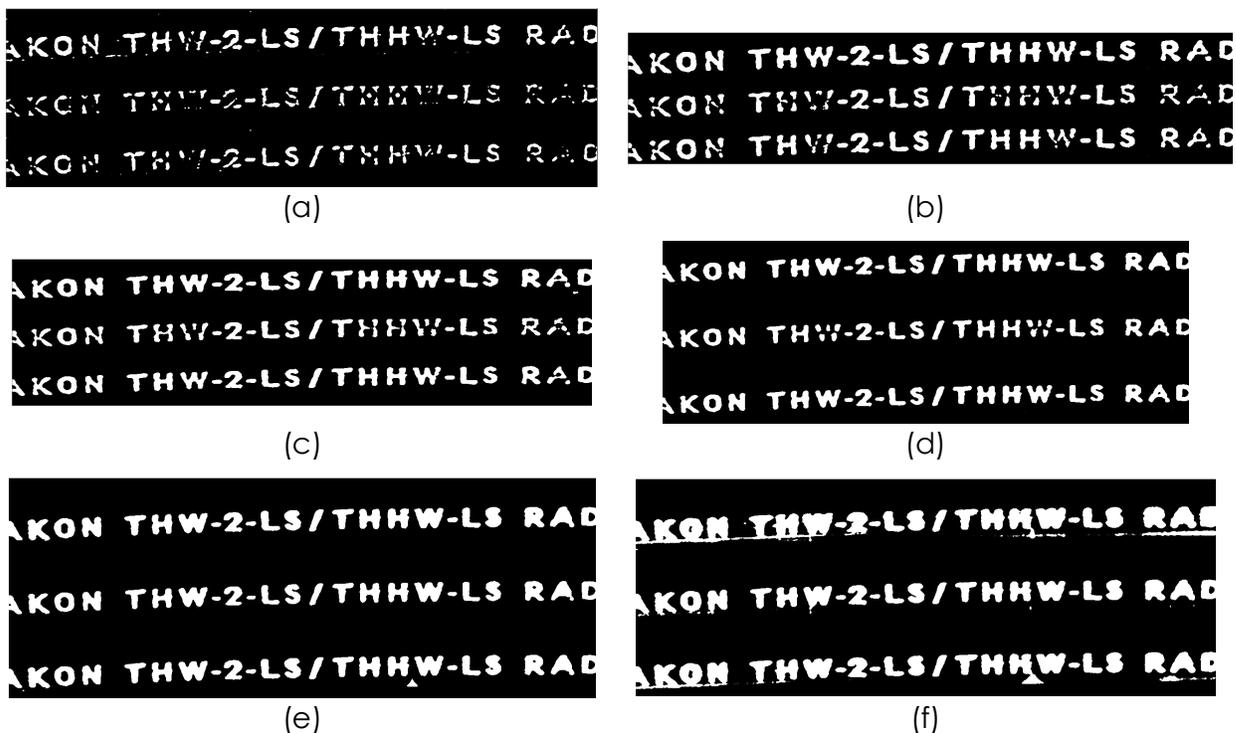


Figura 47. Comparación de distintos elementos estructurales. (a) Rect, Cross y Ellipse 11x11; (b) Rect, Cross y Ellipse 17x17; (c) Rect, Cross y Ellipse 21x21; (d) Rect, Cross y Ellipse 31x31; (e) Rect, Cross y Ellipse 51x51; (f) Rect, Cross y Ellipse 81x81

Fuente: elaboración propia

De la Figura 47 es posible concluir que el tamaño del E. S. define cuáles serán los píxeles que serán incluidos en la operación morfológica. A partir de esta figura, se realizan las siguientes observaciones:

- En (a), al tener un E. S. pequeño, algunos píxeles internos del texto no son incluidos por lo tanto es necesario utilizar un E. S. más grande.
- En (b), en la primera imagen de ejemplo, el E. S. rectangular ya incluye una parte considerable del texto, por lo cual este E. S. podría considerarse como un valor a utilizar. Sin embargo, para los otros dos ejemplos de E. S. aún existen segmentos de texto faltantes.
- En (c), tanto el E. S. rectangular y la elipse podrían considerarse como suficientes, teniendo solamente como detalle la posibilidad de que la letra K en el inicio del texto sea confundida por el motor OCR con una X; el elemento de cruz todavía contiene segmentos faltantes del texto sobre todo en las esquinas

de las letras con componentes verticales y horizontales, en este caso la H y la W, marcando aquí una debilidad de este E. S.

- En (d), los E. S. rectangular y elipse han mejorado el borde del texto, lo cual permite mejorar el resultado del OCR; en cuanto al elemento de cruz todavía se tienen segmentos faltantes, pero ahora corresponden solamente a segmentos de la letra W.
- En (e), los tres E. S. podrían considerarse satisfactorios, ya que en cada uno el grosor del texto es de buen tamaño, pero al tener un E. S. de 51x51, los huecos de algunas letras pueden desaparecer. Tal es el caso del hueco de la penúltima letra, en donde ha desaparecido el hueco de la A. De igual manera, en el elemento elipse, se creó un triángulo entre la H y la W, debido que pueden existir pixeles con valores altos de intensidad; los cuales empiezan a generar ruido en la imagen.
- En (f), los tres E. S. al tener un tamaño grande generan un daño o corrupción en el texto, debido a que se empiezan a considerar pixeles que corresponden a ruido o los pixeles empiezan a verse afectados por las letras vecinas. En el E. S. rectangular, el ruido generado tiene la característica de ser horizontal. En el E. S. cruz, el ruido se genera en los bordes de las letras o la parte inferior a estas. En el E. S. elipse se generan pequeños triángulos o líneas horizontales en la parte inferior de las letras. Cada una de estas situaciones se debe a la forma del E. S.

Del primer experimento es posible concluir que el tamaño dependerá del elemento estructural. Para el E. S. rectangular, el rango de valores a utilizar está entre 17 y 51 unidades. Para el E. S. cruz, el rango de valores a utilizar está entre 31 y 51 unidades. Para el E. S. elipse, el rango de valores a utilizar está entre 17 y 51 unidades.

## **Experimento 2**

El segundo experimento consistió en verificar si los tamaños de E. S. obtenidos en el Experimento 1 se pueden utilizar para los distintos cables. La Figura 48 presenta los resultados obtenidos del Experimento 2.

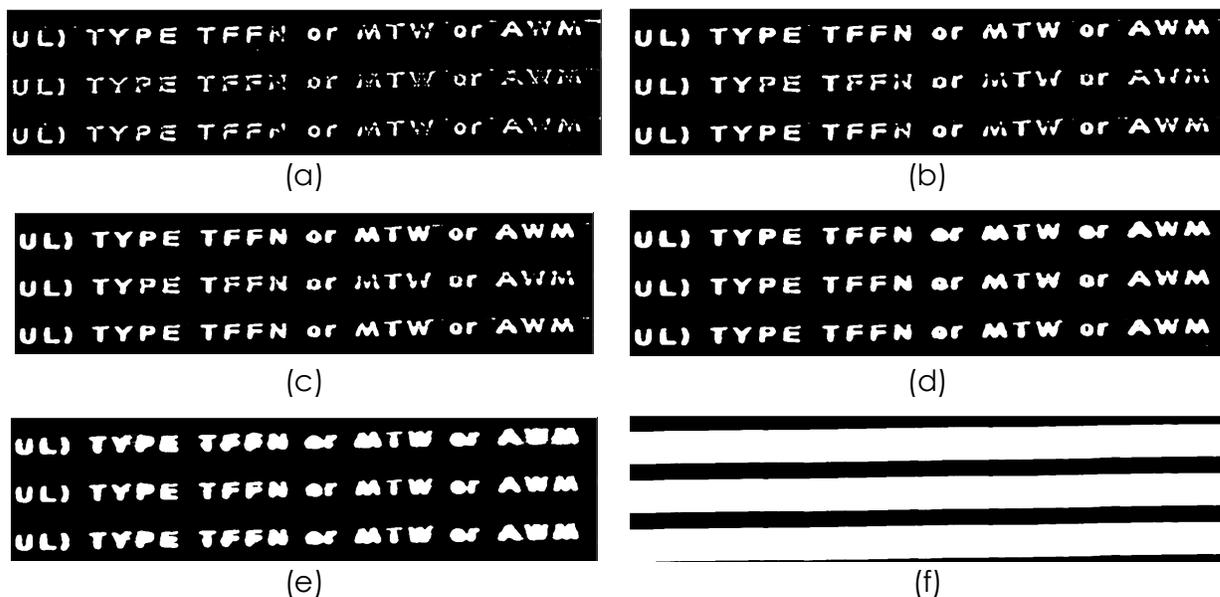


Figura 48. Comparación de distintos elementos estructurales. (a) Rect, Cross y Ellipse 11x11; (b) Rect, Cross y Ellipse 17x17; (c) Rect, Cross y Ellipse 21x21; (d) Rect, Cross y Ellipse 31x31; (e) Rect, Cross y Ellipse 51x51; (f) Rect, Cross y Ellipse 81x81

Fuente: elaboración propia

En la Figura 48 se presenta la problemática obtenida al aplicar el método TBH con los E. S. anteriormente analizados en una imagen de la colección Viakon UL. A partir de esta figura, se realizan las siguientes observaciones:

- En (a), al ser los tres E. S. de tamaño pequeño, el texto presenta secciones faltantes, pero a la vez se empiezan a presentar píxeles cuyo valor de intensidad generan ruido en la imagen.
- En (b), el ruido presentado anteriormente es disminuido, pero aún se tienen segmentos de letras faltantes en los E. S. cruz y elipse. El E. S. rectangular podría considerarse útil.
- En (c), los píxeles correspondientes al ruido se han unido en el E. S. rectangular, pero sin unirse al texto. En el E. S. elipse los píxeles de ruido han generado pequeños círculos. En el E. S. cruz aún existen segmentos del texto faltantes.
- En (d), los píxeles correspondientes al ruido han desaparecido. En el E. S. rectangular desafortunadamente se han unido las letras O y R, lo cual generará errores en el OCR. Los otros dos E. S. se pueden considerar útiles.

- En (e), en los tres E. S. el espacio entre las letras empieza a cerrarse. Esto puede generar problemas para el OCR ya que las características del texto se empiezan a perder. A su vez se empiezan a unir más las letras cercanas.
- En (f), los píxeles que corresponden al texto se han corrompido y mezclado con el cable, por lo cual se ha formado un rectángulo solamente, sin la posibilidad de extraer el texto.

Del segundo experimento es posible concluir que desafortunadamente los E. S. básicos incluidos en OpenCV podrían utilizarse solamente en ciertos casos, sin importar el tamaño del elemento estructural. En esta tesis se encontró una combinación de los elementos estructurales básicos que funcionan para todos los casos a estudiar.

### Experimento 3

Para resolver el problema obtenido en el segundo experimento se propuso un elemento estructural formado por la Ecuación 6 presentada en la Sección 4.3.1. En este experimento se realizó el mismo análisis de los otros dos experimentos, pero utilizando el elemento estructural propuesto. En la Figura 49 se presenta el resultado de aplicar el E. S. propuesto en el método TBH.



**UL) TYPE TFFN OF MTW OF AWM**

(f)

Figura 49. Comparación del elemento estructural propuesto. (a) E. S. propuesto 11x11; (b) E. S. propuesto 17x17; (c) E. S. propuesto 21x21; (d) E. S. propuesto 31x31; (e) E. S. propuesto 51x51; (f) E. S. propuesto 81x81

Fuente: elaboración propia

A partir de la Figura 49, se realizan las siguientes observaciones:

- En (a), el resultado de utilizar un kernel de tamaño pequeño, es similar al de los E. S. básicos; el texto contiene muchos segmentos faltantes, por lo cual no se puede utilizar un tamaño de kernel pequeño.
- En (b), es posible observar cómo el texto contiene una línea horizontal faltante; el texto aún se encuentra muy segmentado.
- En (c), las líneas horizontales de las letras T y F no se han incluido en el resultado. Esta característica se presenta por la forma con la cual se diseñó el E. S. propuesto.
- En (d), algunas letras empiezan a estar completas, por ejemplo: la palabra or y UL); a su vez, los elementos horizontales en las letras F empiezan a tomar forma.
- En (e), el texto segmentado se puede considerar útil ya que no contiene segmentos faltantes y los elementos horizontales de las imágenes anteriores ya son incluidos. Este puede considerarse como un tamaño inicial para el E. S. propuesto.
- En (f), algunos espacios internos de las letras se han cerrado, debido a la proximidad y el tamaño del E. S., pero el texto se mantiene sin deformaciones. Este tamaño del E. S. se puede considerar como el límite superior ya que ha aparecido una pequeña línea horizontal en la parte inferior de la antepenúltima letra A; por lo cual si se aumenta el tamaño del E. S. se pueden presentar nuevamente los problemas obtenidos en el segundo experimento.

Del experimento 3 se puede concluir que el E. S. propuesto para el método TBH permite segmentar el texto de las imágenes. La Tabla 7 presenta una comparación del método TBH utilizando los límites inferiores y superiores para el E. S. propuesto.

Tabla 7. Aciertos por conjunto utilizando los límites para el elemento estructural propuesto

Conjunto	Tamaño del E. S.		Aciertos / Conjunto
	51x51	81x81	
<b>Indiana rojo</b>	78.5	<b>82</b>	91.00
<b>Viakon blanco</b>	44.058	<b>48.411</b>	50.00
<b>Kobrex azul</b>	59.555	<b>60.333</b>	83.00
<b>Kobrex rojo</b>	36.5	<b>37.555</b>	41.00
<b>Kobrex Vinikob</b>	65.09	<b>68.09</b>	83.00
<b>Viakon negro</b>	78.571	<b>79.5</b>	87.00
<b>Viakon UL</b>	<b>57</b>	56.5	68.00
<b>Viakon verde</b>	81.954	<b>83.09</b>	87.00
<b>Sumatoria</b>	501.228	<b>515.479</b>	590.00

De la Tabla 7 se toma la sumatoria de los aciertos de letras reconocidas por tamaño de E. S. El E. S. con mayor sumatoria es el utilizado en el método TBH.

### 5.3.3 Número de clusters para método propuesto KMN

El número de clusters define cuantos grupos diferentes se van a generar dependiendo del valor de los pixeles; para elegir el número correcto de clusters se realizó el siguiente análisis:

#### Tres clusters

Al elegir tres clusters se busca que cada uno corresponda al fondo de la imagen, al cable y a la etiqueta en el cable, respectivamente; pero al realizar este experimento, existen cables en donde el valor de los pixeles que corresponden al cable y los que corresponden al texto tienen una diferencia de valor pequeña, por lo cual estos valores se agrupan en el mismo cluster; un ejemplo de esto se presenta en la Figura 50.

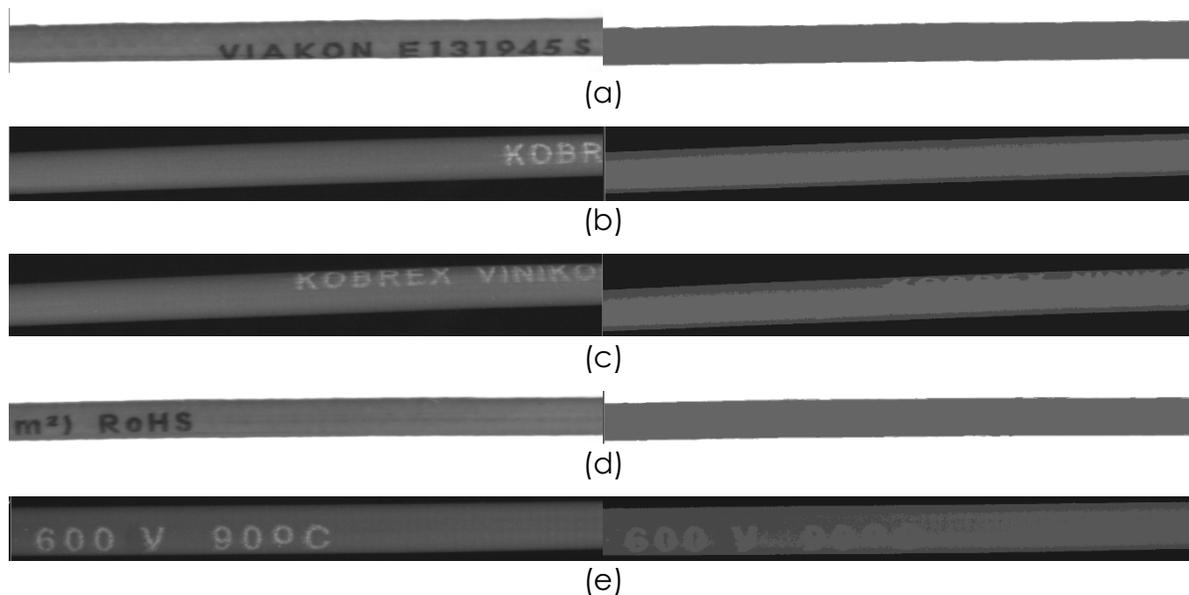


Figura 50. Errores al utilizar 3 clusters en el método KMN dentro de distintas colecciones.

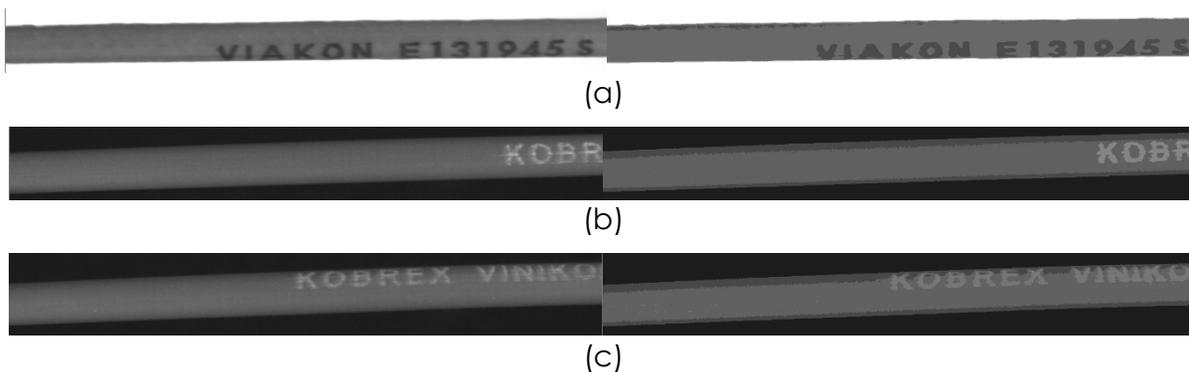
(a) Viakon UL; (b) Kobrex Azul; (c) Kobrex Vinikob; (d) Viakon UL; (e) Kobrex Rojo

Fuente: elaboración propia

En la Figura 50 es posible observar cómo se han agrupado de manera errónea los píxeles correspondientes al texto y algunos píxeles correspondientes al cable, por lo cual la segmentación del texto contendrá píxeles que no corresponden al texto de la etiqueta; esto afecta directamente la posibilidad del sistema de detectar falsos negativos.

### Cuatro clusters

Al elegir cuatro clusters se busca mejorar el resultado obtenido al elegir tres clusters; añadiendo un cuarto cluster se busca que este cluster contenga los valores que son clasificados erróneamente en el experimento anterior. La Figura 51 presenta un ejemplo utilizando la misma imagen del experimento anterior, pero con un cluster más.



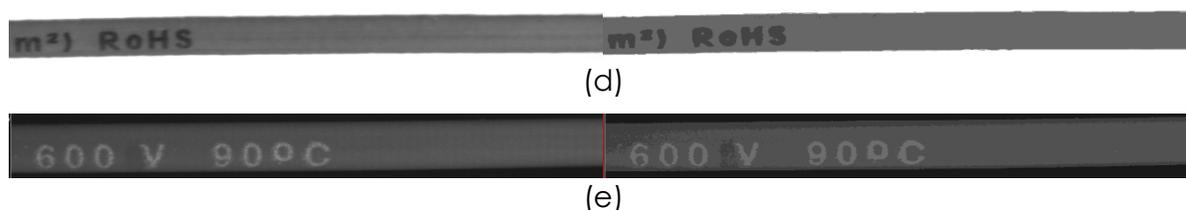


Figura 51. Resultado al utilizar 4 clusters con el método KMN dentro de distintas colecciones. (a) Viakon UL; (b) Kobrex Azul; (c) Kobrex Vinikob; (d) Viakon UL; (e) Kobrex Rojo

Fuente: elaboración propia

En la Figura 51 se presentan los mismos cables de la Figura 50, pero con la diferencia de que se utilizaron cuatro clusters en el método KMN. De esta manera los pixeles que se clasifican erróneamente al utilizar 3 clusters pueden ser clasificados de manera correcta; ya que este cluster extra permite asociar a este grupo los pixeles con valor intermedio entre el cable y el texto. Por lo tanto, para disminuir la posibilidad de que el sistema genere falsos negativos se justifica la utilización de cuatro clusters en el método KMN.

### 5.3.4 Elección de reglas en WEKA para el algoritmo propuesto S.A.F.E.

Para el algoritmo S.A.F.E. se decidió utilizar la herramienta WEKA [81]; esta contiene algunos algoritmos de clasificación basados en reglas. El algoritmo S.A.F.E. permitió utilizar las reglas de decisión para clasificar errores y regiones con ruido en el dataset empleado en esta tesis. Para seleccionar el clasificador a utilizar se generó la Tabla 8, la cual fue obtenida evaluando los clasificadores JRIP[82], DecisionTable[82], OneR[82], PART[82] y Ridor[82] con 2023 instancias, cada una con 7 atributos.

Tabla 8. Comparación entre clasificadores basados en reglas

Clasificador	Precisión	Exhaustividad	F-Measure	Número de reglas
JRIP	0.985	0.986	0.986	5
DecisionTable	0.975	0.975	0.975	48
OneR	0.959	0.956	0.957	7
PART	<b>0.988</b>	<b>0.988</b>	<b>0.988</b>	10
Ridor	0.983	0.983	0.983	11

En la Tabla 8 el clasificador con mejor precisión, exhaustividad y F-measure es el clasificador PART, pero genera 10 reglas. El segundo mejor clasificador es el clasificador JRIP; este tiene una diferencia contra el primero de tan solo 0.003 en precisión, exhaustividad y F-measure, pero genera solamente cinco reglas de decisión; lo que significa que con la mitad de las reglas del clasificador PART se puede obtener una precisión muy similar utilizando las reglas de JRIP. Por lo tanto, el clasificador seleccionado en esta tesis para ser usado por el algoritmo S.A.F.E. es el de JRIP.

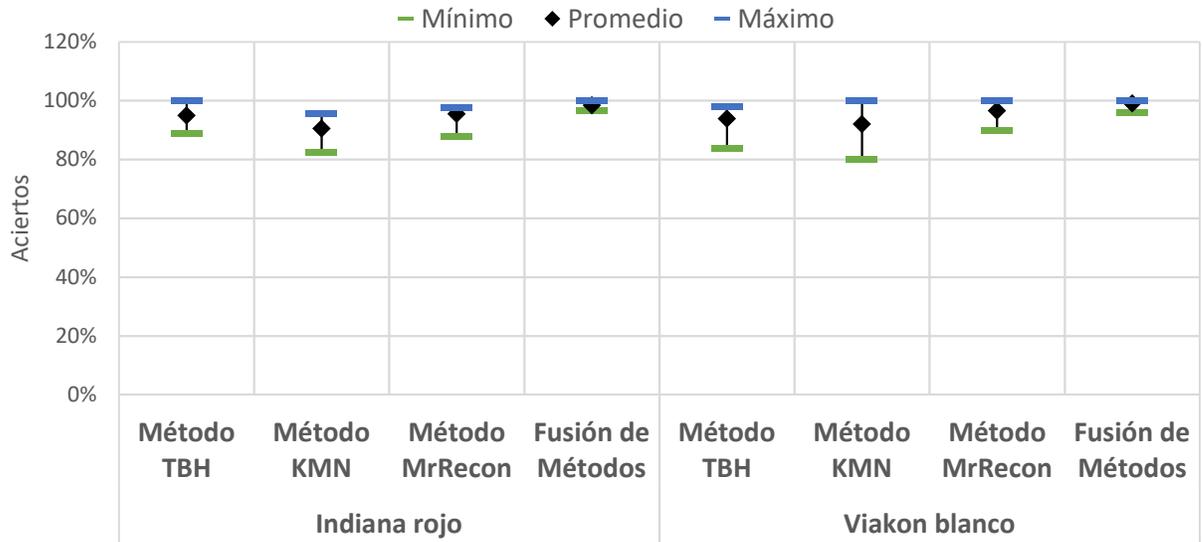
#### **5.4 EVALUACIÓN DEL DESEMPEÑO DEL SISTEMA**

Para evaluar el desempeño del sistema se obtuvieron los aciertos por colección y método. En la Tabla 9 se presentan los resultados de esta evaluación y en la Figura 52 se presentan gráficamente estos resultados.

Tabla 9. Resultados de la evaluación del desempeño del sistema por colección y método, correspondiendo cada resultado al porcentaje de aciertos obtenidos.

	Indiana rojo				Viakon blanco			
Valor	Método TBH	Método KMN	Método MrRecon	Fusión de Métodos	Método TBH	Método KMN	Método MrRecon	Fusión de Métodos
Mín.	89%	82%	88%	97%	84%	80%	90%	96%
Prom.	95%	91%	96%	98%	94%	92%	97%	99%
Máx.	100%	96%	98%	100%	98%	100%	100%	100%
	Kobrex azul				Kobrex rojo			
Valor	Método TBH	Método KMN	Método MrRecon	Fusión de Métodos	Método TBH	Método KMN	Método MrRecon	Fusión de Métodos
Mín.	7%	23%	54%	61%	5%	83%	71%	88%
Prom.	47%	48%	73%	79%	41%	91%	91%	94%
Máx.	86%	72%	95%	96%	83%	98%	98%	98%
	Kobrex Vinikob				Viakon negro			
Valor	Método TBH	Método KMN	Método MrRecon	Fusión de Métodos	Método TBH	Método KMN	Método MrRecon	Fusión de Métodos
Mín.	0%	30%	40%	59%	79%	77%	68%	89%
Prom.	30%	67%	65%	84%	90%	87%	88%	92%
Máx.	63%	90%	88%	99%	99%	97%	98%	95%
	Viakon UL				Viakon verde			
Valor	Método TBH	Método KMN	Método MrRecon	Fusión de Métodos	Método TBH	Método KMN	Método MrRecon	Fusión de Métodos
Mín.	18%	26%	16%	26%	87%	72%	76%	91%
Prom.	84%	80%	83%	88%	96%	95%	95%	99%
Máx.	94%	99%	97%	100%	99%	99%	99%	100%

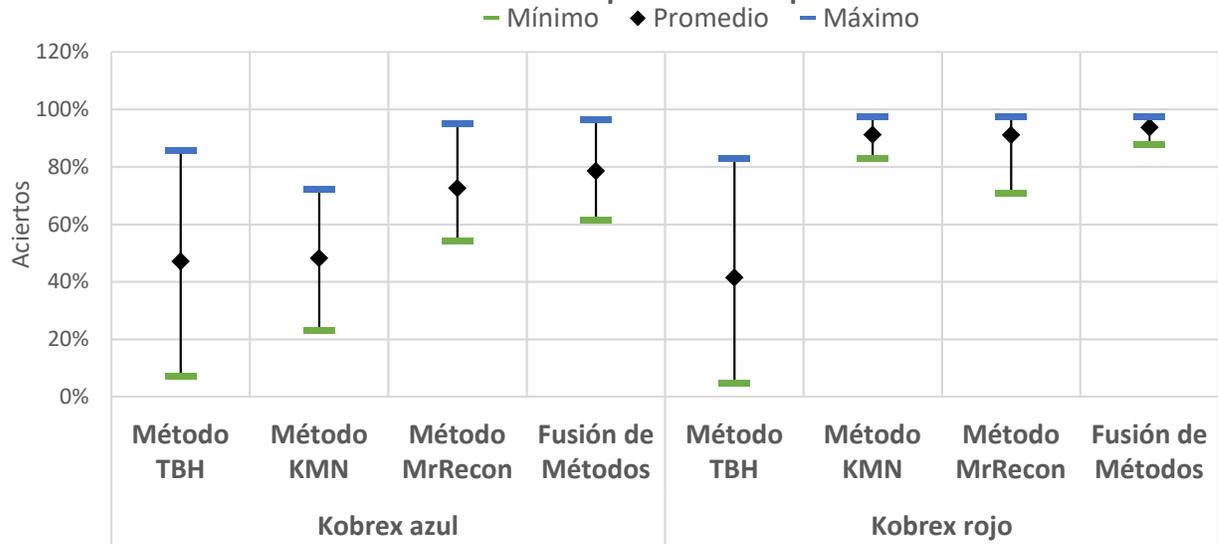
## Verdaderos positivos por colección



(a)

(b)

## Verdaderos positivos por colección



(c)

(d)

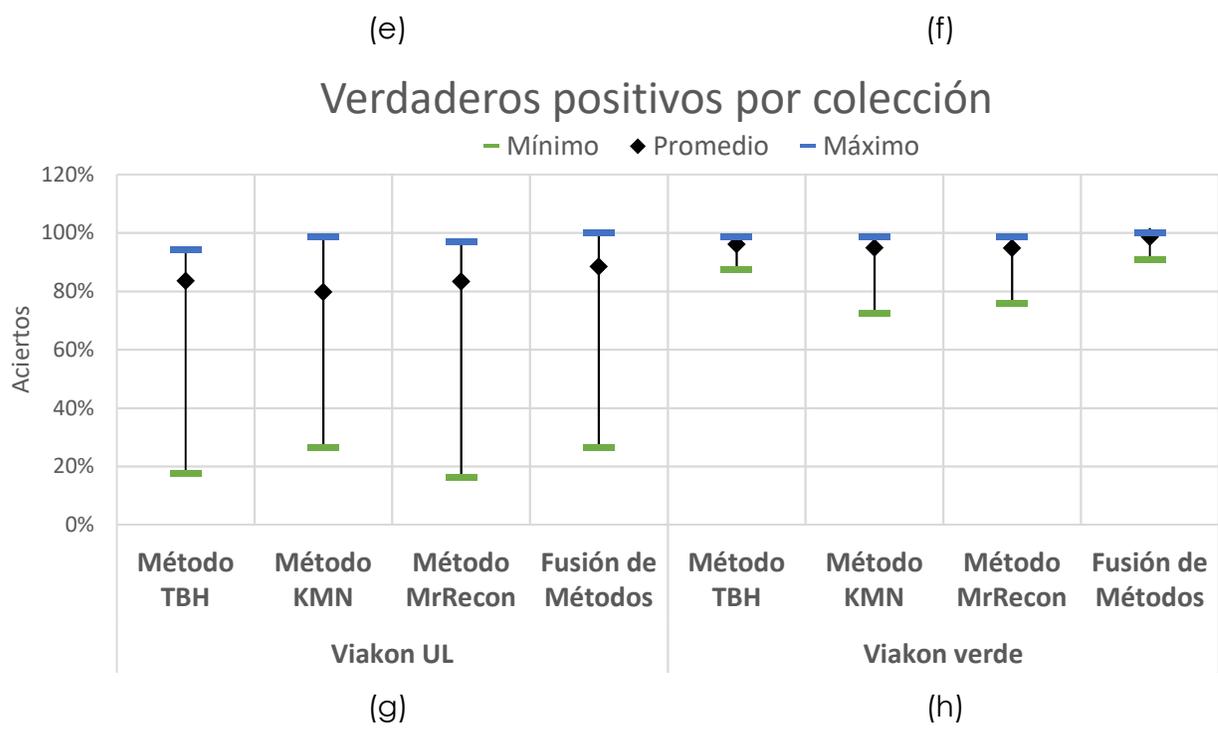
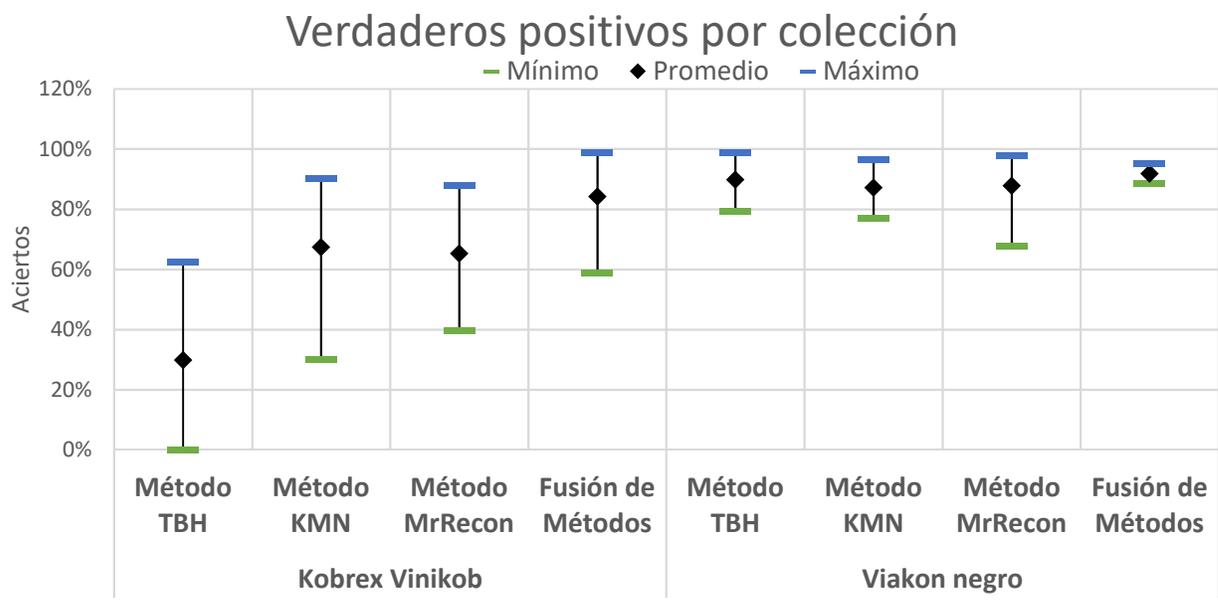


Figura 52. Aciertos por colección de imágenes, evaluando cada uno de los métodos propuestos y su combinación

Fuente: elaboración propia

En la Figura 52 se presentan los aciertos o verdaderos positivos obtenidos al evaluar los diferentes grupos de imágenes que forman el dataset desarrollado. En esta figura se presentan tres indicadores para cada método; en color azul, el valor máximo en aciertos;

en color negro y en forma de rombo se indica el promedio de aciertos; y en color verde se presenta el número mínimo de aciertos obtenidos.

A partir de la Figura 52 se realizan las siguientes observaciones:

- En (a), en la colección correspondiente a Indiana rojo, los métodos propuestos se desempeñan de buena manera, ya que todos los porcentajes mínimos de aciertos se encuentran por encima del 90%, obteniendo incluso un desempeño mayor al 95% al fusionar la validación de los métodos.
- En (b), en la colección de Viakon blanco, el desempeño de los métodos individuales se considera buena ya que también presenta como mínimo un 80% de aciertos, obteniendo un desempeño por arriba del 90% al fusionar la validación de los métodos. El método que más variación presenta es el de KMN; esto se debe a que al tener un alto contraste el cable y el texto en las imágenes, los pixeles con poca diferencia de valor en su intensidad se pueden llegar a clasificar en diferentes grupos. Este problema se presenta principalmente en los pixeles que corresponden al borde del texto, los cuales por lo general son de menor intensidad que en el cuerpo de cada uno de los caracteres en el cable.
- En (c), la colección Kobrex azul, como se mencionó en la Sección 5.4, contiene una impresión de texto deficiente. En el método de TBH, el valor mínimo de aciertos se encuentra por debajo del 10%. Sin embargo, el valor máximo se encuentra arriba del 80%, por lo cual este método presenta una alta variación en sus resultados. En el caso del método de KMN, el mínimo se encuentra por arriba del 20%, pero su máximo no alcanza a superar el 80%, por lo cual, para este tipo de cable, este método no sería el indicado individualmente. El método de MrRecon es el que más aciertos tiene, con un promedio del 73%. Sin embargo, esto es un número bajo para poder validar correctamente el texto en este producto. La fusión de la validación de los métodos en promedio obtiene casi un 80% de aciertos con una variación entre el 61% y el 96%, demostrando la ventaja de utilizar la fusión de los métodos, a pesar de que el método TBH y KMN contienen una alta variación esta es compensada al utilizar la validación resultante de los tres métodos.

- En (d), en la colección Kobrex rojo se presenta una situación similar que en el grupo anterior con el método de TBH. Sin embargo, en este producto los métodos de KMN y MrRecon obtienen un mayor porcentaje de aciertos. KMN obtuvo un promedio del 91% de aciertos, y una variación entre el 83% y el 98%. El método de MrRecon obtiene un promedio del 91% de aciertos con una variación entre el 71% y 98%. Esta variación se origina porque algunos píxeles del texto no son reconstruidos por el método MrRecon, lo cual disminuye la cantidad de aciertos. Finalmente, la fusión de la validación de los métodos obtiene un promedio de aciertos del 94% con una variación entre el 88% y 98%.
- En (e), en la colección de Kobrex Vinikob en el método de TBH, el promedio de aciertos es del 30% con una variación entre el 0% y el 63%; esto es debido a que el producto contiene características en las cuales el método de TBH no se puede desarrollar de manera correcta. El método de KMN obtiene un promedio de aciertos del 67% con una variación entre el 30% y el 90%; esta variación se origina porque algunos componentes o partes de las letras del etiquetado contienen valores cuya diferencia son clasificados en grupos distintos, generando así una detección errónea en el OCR. El método de MrRecon obtiene un promedio de aciertos del 65% con una variación entre el 40% y el 88%; esta debido al texto con segmentos borrosos que al ser reconstruidos generan ruido en el texto segmentado. El fusionar la validación de los métodos anteriores permite aumentar el porcentaje de aciertos al 84% con una variación del 59% al 99%.
- En (f), en la colección de Viakon negro, todos los métodos obtienen un buen porcentaje de aciertos; el método de TBH obtiene un promedio de aciertos del 90% con una variación del 79% al 99%; por lo tanto, este método se desempeña bien en situaciones de alto contraste entre el cable y el texto. El método de KMN obtuvo un promedio de aciertos del 87% con una variación del 77% al 97%. Esto significa que el texto impreso en este producto contiene algunos píxeles con un valor de intensidad menor en el cuerpo del carácter o en los bordes del mismo, pero a pesar de esto, el método permite obtener un buen porcentaje de aciertos. El método de MrRecon obtuvo un desempeño similar al del método de KMN, pero tiene una variación mayor. En la fusión de la validación

de los métodos el valor máximo se encuentra ligeramente por abajo del 100%, pero al presentar un promedio de aciertos del 92%, con una variación entre el 89% y el 95%, puede significar que algunos elementos del texto impreso contienen defectos o daño en la impresión.

- En (g), la colección de Viakon UL contiene una alta variación en el porcentaje de aciertos de todos los métodos, incluso en la fusión de las validaciones. En los cuatro métodos, el porcentaje de validaciones se encuentra por encima del 80%. Sin embargo, todos los métodos tienen un mínimo menor al 30%. Esto debido a que la colección contiene imágenes con características que dificultan la detección correcta del texto en el cable. Pero también se presentan máximos mayores al 93% por lo cual se presentan casos en donde se validaron todos los caracteres del texto.
- En (h), en la colección de Viakon verde, los tres métodos y su fusión obtienen un promedio de aciertos del 95%, lo cual significa que los métodos permiten validar el texto con una sola secuencia de imágenes.

A partir de los aciertos o verdaderos positivos por colección se calculó el error en las colecciones y se determinó un error global. La Tabla 10 presenta los resultados de dicho análisis.

Tabla 10. Porcentaje de error en las distintas colecciones

Núm. de carácter es por etiqueta	Conjunto	% Error				
		Original sin procesamiento	Método TBH	Método KMN	Método MrRecon	Fusión de métodos
<b>91.00</b>	<b>Indiana rojo</b>	100.00%	4.51%	8.68%	4.17%	<b>1.51%</b>
<b>50.00</b>	<b>Viakon blanco</b>	7.06%	6.00%	7.88%	3.29%	<b>0.82%</b>
<b>83.00</b>	<b>Kobrex azul</b>	98.80%	52.88%	51.81%	27.44%	<b>21.42%</b>
<b>41.00</b>	<b>Kobrex rojo</b>	100.00%	58.54%	8.81%	8.94%	<b>6.23%</b>
<b>83.00</b>	<b>Kobrex Vinikob</b>	99.89%	70.10%	32.53%	34.61%	<b>15.66%</b>
<b>87.00</b>	<b>Viakon negro</b>	55.17%	10.10%	12.73%	12.15%	<b>8.13%</b>
<b>68.00</b>	<b>Viakon UL</b>	100.00%	16.42%	20.22%	16.67%	<b>11.52%</b>
<b>87.00</b>	<b>Viakon verde</b>	100.00%	3.92%	5.12%	5.17%	<b>1.41%</b>
	Promedio	<b>82.61%</b>	<b>27.81%</b>	<b>18.47%</b>	<b>14.06%</b>	<b>8.34%</b>
	Promedio ponderado	<b>85.33%</b>	<b>26.53%</b>	<b>19.45%</b>	<b>14.75%</b>	<b>8.69%</b>

La Tabla 10 presenta el cálculo del porcentaje de error por conjunto o colección. De esta tabla es posible observar que la etapa de pre procesamiento, sin importar el método que se utilice, disminuye el porcentaje de error; excepto en el producto de Viakon blanco, donde el contraste entre el color del cable y del texto permite obtener un error del 7.06% sin requerir procesamiento en la imagen. Sin embargo, en los otros productos se presentan casos con error del 100%. Asimismo, es posible observar que todos los métodos tienen dificultades con los cables de los productos Kobrex, debido a las características que presenta su etiquetado. En cuanto a los productos de Viakon e Indiana, los tres métodos tienen resultados diferentes ya que cada uno tiene sus ventajas y desventajas ante los distintos problemas presentados en el dataset (Sección 5.1).

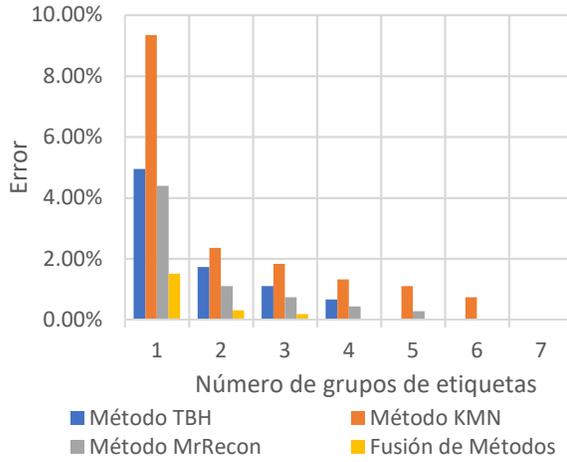
De la Tabla 10 es posible concluir que la mejor manera de disminuir el porcentaje de error y volver más robusto al sistema ante los diferentes productos a trabajar se logra combinando la validación de cada uno de los métodos desarrollados. Lo cual permite complementar la validación de unos con otros, para así poder realizar una validación precisa. El porcentaje de error global al utilizar el método de fusión permite obtener un error del 8.69% global.

## **5.5 ANÁLISIS PARA DETERMINAR EL NÚMERO DE SECUENCIAS NECESARIAS PARA LA VALIDACIÓN DE ETIQUETAS POR EL ALGORITMO VALET**

Elegir el número de secuencias necesarias para la validación de etiquetas le permite al sistema evitar generar falsos negativos en los caracteres validados por el algoritmo VALET. Una secuencia de imágenes es un número  $N$  de imágenes las cuales contienen un segmento del texto; los segmentos de texto de las imágenes de la secuencia en conjunto forman la etiqueta completa del cable.

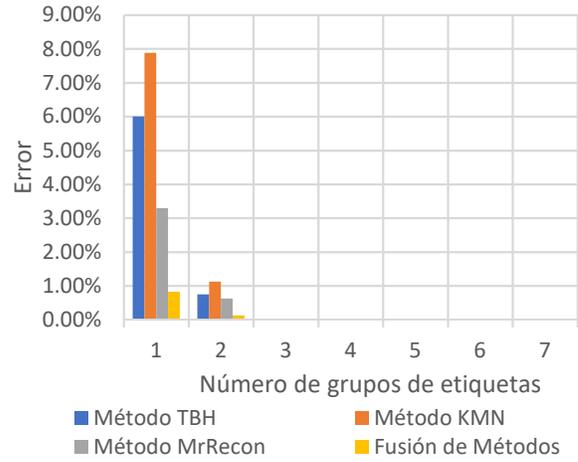
Para determinar el número de secuencias que permite evitar generar falsos positivos y disminuir la tasa de error del sistema, se realizó un análisis combinando de una a siete secuencias. Al utilizar más secuencias en la validación del texto se espera observar una disminución en el porcentaje de error. La Figura 53 presenta los resultados por grupo de cable de este análisis.

Validación de etiquetas en grupos de secuencias - Indiana rojo



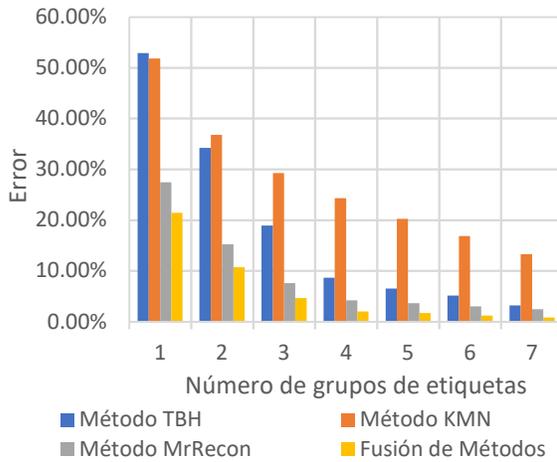
(a)

Validación de etiquetas en grupos de secuencias - Viakon blanco



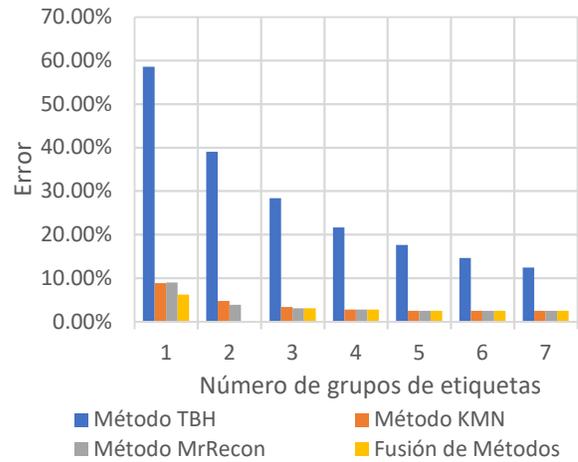
(b)

Validación de etiquetas en grupos de secuencias - Kobrex azul



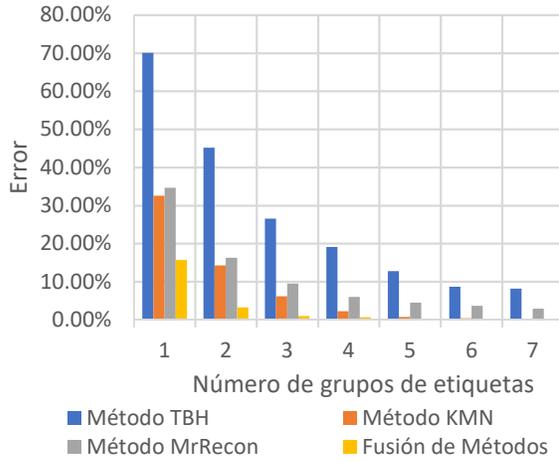
(c)

Validación de etiquetas en grupos de secuencias - Kobrex rojo



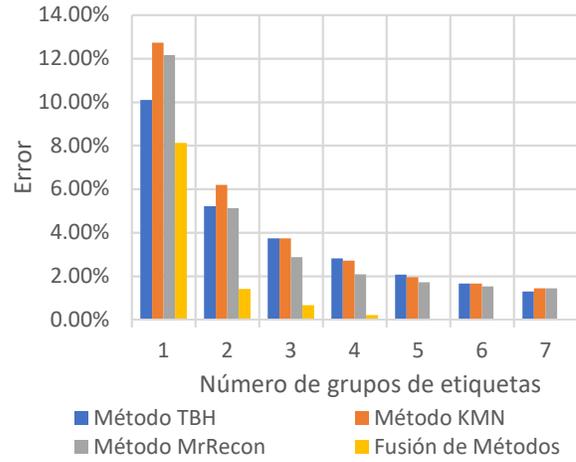
(d)

Validación de etiquetas en grupos de secuencias - Kobrex Vinikob



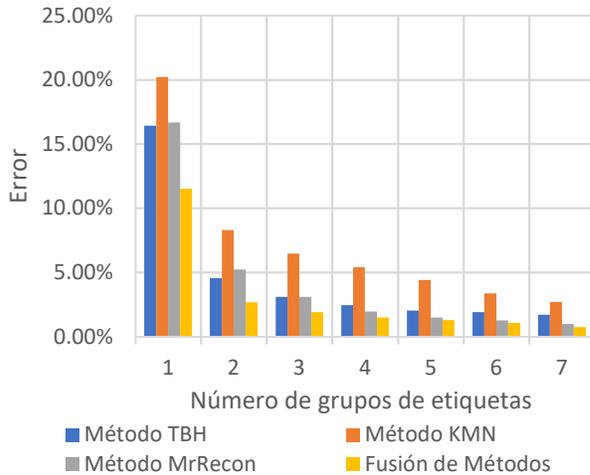
(e)

Validación de etiquetas en grupos de secuencias - Viakon negro



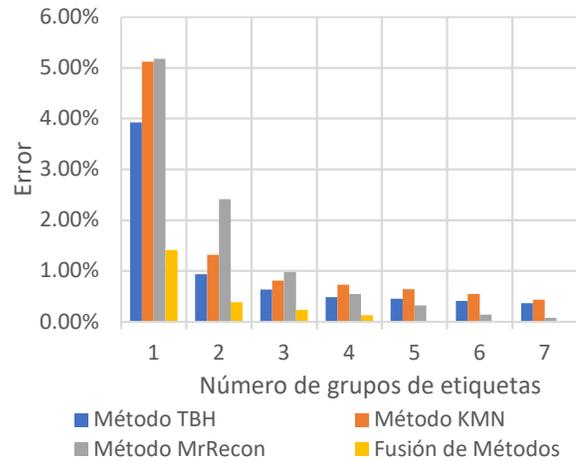
(f)

Validación de etiquetas en grupos de secuencias - Viakon UL



(g)

Validación de etiquetas en grupos de secuencias - Viakon verde



(h)

Figura 53. Validación de etiquetas por número de secuencias, comparando el número de secuencias contra el porcentaje de error. (a) corresponde al grupo de imágenes del cable Indiana rojo; (b) corresponde al grupo de imágenes del cable Viakon blanco; (c) corresponde al grupo de imágenes del cable Kobrex Azul; (d) corresponde al grupo de imágenes del cable Kobrex rojo; (e) corresponde al grupo de imágenes del cable Kobrex Vinikob; (f) corresponde al grupo de imágenes del cable Viakon negro; (g) corresponde al grupo de imágenes del cable Viakon UL; y (h) corresponde al grupo de imágenes del cable Viakon verde Fuente: elaboración propia

A partir de la Figura 53 se realizan las siguientes observaciones:

- En (a), el número necesario para disminuir el error al 0% es de cuatro secuencias para la validación fusionada de los métodos. Sin embargo, para poder disminuir todos los métodos al 0% de error, el número de secuencias sube a 7. Por lo tanto, para el cable Indiana rojo se necesitarían entre 4 y 7 secuencias para disminuir el error en la validación. En el método de KMN la tasa de disminución por número de secuencias es muy baja y esto se debe a que este método es susceptible a píxeles con valores de intensidad cercanos, por lo cual se pueden llegar a clasificar erróneamente los píxeles o incluir ruido en el texto segmentado.
- En (b), el número de secuencias necesarias para disminuir el error al 0% de todos los métodos es de 3 grupos. Este comportamiento se debe a que en el cable Viakon blanco el contraste entre el cable y el texto es muy alto, permitiendo así disminuir el número de secuencias necesarias para poder validar la etiqueta de manera correcta.
- En (c), se empieza con un error elevado para todos los métodos. Esto se debe a las características de este grupo; el cable Kobrex azul contiene un texto impreso con segmentos borrosos en las letras, lo cual causa problemas en la etapa del OCR a pesar de tener una etapa de pre procesamiento. Por lo tanto, a pesar de utilizar 7 secuencias, el error se vuelve pequeño en el caso de la fusión de métodos, excepto en el método KMN, en el cual las partes borrosas del texto extraen componentes que le permiten al motor OCR detectar de manera correcta el texto impreso.
- En (d), el método de TBH empieza con un error cercano al 60%; esto se debe a que el método es susceptible a errores cuando el texto impreso no tiene un alto contraste y llega a contener píxeles del mismo color del cable en el interior del texto. Sin embargo, esto no sucede con los otros métodos, por lo cual, a pesar de utilizar 7 secuencias para validar el texto, el error se mantiene por encima del 10% para el método de TBH; en los otros métodos, después de utilizar 4 secuencias, ocurre una situación que no se había presentado; el porcentaje de error no ha disminuido a pesar de utilizar más secuencias. Esto significa que

el texto que se valida contiene letras con daño o problemas de impresión; ya que ninguno de los métodos, ni la fusión de ellos, disminuye el error.

- En (e), se presenta un caso similar al de (d) en donde el texto contiene muchas imperfecciones desde el etiquetado, por lo cual el método TBH tiende a equivocarse. Sin embargo, cuando se utiliza una sola secuencia de imágenes, los otros métodos también presentan un error alto. En este caso, al utilizar 5 secuencias y fusionar los métodos, se disminuye el error a 0%. Esto se debe a que el método KMN, también contiene un error pequeño.
- En (f), el texto presenta detalles en la impresión como segmentos de texto tallados o faltantes gotas de tinta y texto borroso. En la gráfica obtenida se aprecia que, aunque se utilice la fusión de los métodos y 5 secuencias, el error en los métodos individuales no disminuye a 0%, y tiende a disminuir con una tasa menor conforme se van utilizando más secuencias, permitiendo validar nuevamente la manera de detectar que existen problemas con el texto impreso.
- En (g), conforme se van utilizando más secuencias en el cable, el error tiende a disminuir, pero ni tampoco el fusionar los métodos, permite obtener un 0% de error. Esto se debe a que el producto VIAKON UL es uno de los productos considerados como difíciles, que contiene problemas como texto mal alineado y un material con alta reflectividad por lo cual el borde del cable se ve oscurecido lo cual puede confundirse como texto impreso.
- En (h), conforme se van utilizando más secuencias, el error tiene una tendencia descendente, llegando hasta un 0% al utilizar la fusión de métodos con 5 secuencias. En este conjunto de imágenes, el método individual al que más le beneficia el utilizar más secuencias es el método de MrRecon, ya que conforme se aumenta este número, el error disminuye casi hasta llegar el 0%.

También, a partir de la Figura 53, se realizan las siguientes observaciones relacionadas con el comportamiento de los métodos propuestos:

- Método TBH. Este método funciona bien para aquellos cables en donde el texto contiene buenas características de impresión; ya que, a pesar de haberse

propuesto un elemento estructural, al utilizar una operación morfológica es posible incluir ruido dentro de la segmentación del cable. En los productos en donde el texto es borroso o contienen segmentos faltantes de las letras, el error por lo general es alto; al utilizar más de una secuencia de imágenes el error en este método tiende a disminuir, excepto en los casos problemáticos.

- Método KMN. Este método tiende a tener problemas en los productos donde la etiqueta del cable no está completamente coloreada del color de la tinta; ya que por su principio de funcionamiento puede llegar a agrupar erróneamente píxeles que corresponden al texto como píxeles del cable o del borde de la impresión. Por lo contrario, en los casos donde la impresión tiene un relleno uniforme, KMN permite remover el borde de la impresión si es que el texto se empieza a hacer borroso, asignando estos píxeles como otro grupo distinto al del texto, dependiendo del producto a validar. En KMN, por lo general, a partir de la utilización de dos secuencias el error disminuye significativamente.
- Método MrRecon. Este es de los métodos más robustos; ya que su principio de funcionamiento permite extraer aquellos valores en donde se encuentra el texto. Sin embargo, en los casos difíciles como Viakon Negro, Viakon UL y Viakon Verde, es necesario utilizar más de una secuencia para poder disminuir el error; esto se debe a que, en estos productos, los valores que corresponden a los defectos del cable son reconstruidos si estos contienen un valor de intensidad similar al del texto.
- Fusión de Métodos. Esta disminuye en gran medida la tasa de error en cada uno de los conjuntos de imágenes del dataset; ya que al utilizar la validación de todos los métodos anteriores permite complementar aquellos caracteres que no han sido validados en alguno de los métodos de manera individual; al mismo tiempo, es posible observar un comportamiento interesante, al presentarse productos con problemas como Kobrex rojo, Kobrex azul y Viakon UL; ya que en estos, la tasa de error no disminuye, permitiendo encontrar situaciones en donde el texto de estos productos contiene problemas o defectos de impresión.

El análisis realizado considerando utilizar más de una secuencia permite confirmar que el error puede ser decrementado hasta un porcentaje que permita tener el menor número de falsos negativos y validar de manera certera el etiquetado en el cable. De igual manera, es posible concluir que al combinar o fusionar la validación de los métodos propuestos, es posible obtener un error aún menor, otorgándole mayor robustez al sistema. La fusión de métodos se puede convertir en un sistema de votaciones, en donde al haber más de un método con el mismo error de validación, se pueden detectar también los errores en el etiquetado.

## **5.6 DISCUSIÓN DE RESULTADOS**

En las secciones previas se presentaron observaciones de los experimentos, pruebas o del análisis de resultados de las distintas etapas del sistema de visión propuesto en la tesis. Los experimentos permitieron determinar: el espacio de color a utilizar en los métodos de pre procesamiento; el E. S. y su tamaño para el método TBH; el número de clusters para el método KMN; la elección del clasificador para el algoritmo S.A.F.E.; una evaluación del desempeño del sistema de inspección con la base de datos propuesta; y la diferencia en la validación cuando se utilizan un número mayor de secuencias.

Del análisis de resultados presentado en las Secciones 5.3, 5.4 y 5.5, pueden surgir interrogantes como:

1. ¿Qué método de pre procesamiento es el mejor?

Como se observó en la Sección 5.4 y 5.5, el mejor método es la fusión de los métodos, ya que complementa las debilidades de un método, con los pros de los otros. Sin embargo, de manera individual, MrRecon demostró ser el mejor método.

2. ¿Qué resultado puedo esperar en la validación de la etiqueta de un nuevo producto?

Un resultado similar al obtenido en este trabajo, ya que el sistema fue diseñado para trabajar con distintos colores de cables y de etiquetas, teniendo como única limitante el caso en el que la etiqueta sea del mismo color del cable, o casos donde el método de impresión utilice una fuente punteada para el texto.

### 3. ¿En qué caracteres tiene más fallas el OCR?

Dentro de los caracteres en donde el OCR tiene un mayor número de errores se encuentran el punto (.), la coma (,), el exponente al cuadrado (<sup>2</sup>) y el símbolo de grados (°). En otros casos, las letras que más se confunden son la H con la K, la K con la X, la S con un 5, el 1 con una L; esto se debe al parecido gráfico entre estas letras o números.

A continuación, se presentan los descubrimientos más importantes encontrados con la investigación desarrollada. Representada de forma gráfica, en la Figura 54 se presenta una respuesta al interrogante número 1. Cuando el color se encuentra cercano al verde, la eficacia del método es alta, mientras que, al acercarse al rojo, la eficacia del método es baja.

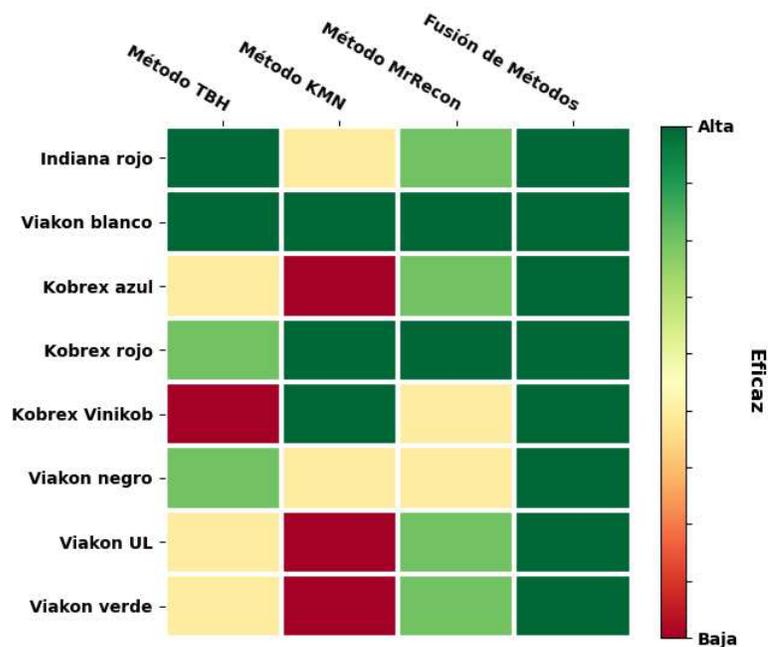


Figura 54 – Nivel de eficacia método versus producto

Fuente: elaboración propia

A partir de la Figura 53 se realizan las siguientes observaciones.

- El método TBH trabaja bien en aquellos productos donde el contraste entre el color del cable y el texto es alto, pero puede presentar problemas en aquellos productos con etiqueta borrosa.

- El método KMN trabaja bien con aquellos productos en donde el texto está borroso, pero empieza a presentar problemas en aquellos productos con poco contraste entre el color del cable y el texto.
- El método MrRecon trabaja bien con la mayoría de los productos, pero presenta un poco de problemas cuando hay defectos en la impresión el texto.
- La fusión de métodos es el método que mejor funciona, ya que los pros y contras de los métodos individuales son compensadas con la fusión de la validación de los métodos, generando una validación con una mayor robustez.

## CONCLUSIONES

El problema estudiado en esta tesis fue la detección de etiquetas en conductores eléctricos recubiertos fabricados en líneas de extrusión, para lo cual se desarrolló un sistema de visión para la detección del texto impreso en los conductores eléctricos, en los cuales el cable y el texto pueden ser de color distinto. El sistema permite validar el texto impreso, identificar y señalar áreas con problemas o defectos de impresión; permitiendo al operador identificar visualmente el resultado de la detección del texto e identificar los elemento o áreas con problemas.

El sistema de visión está formado por distintas etapas: la etapa de captura de imágenes, la etapa de pre procesado, la etapa de detección de ruido y defectos en el texto, la etapa de OCR, la etapa de validación del texto y la etapa de visualización.

En la etapa de captura de imágenes se integró un módulo de adquisición de imágenes con iluminación uniforme, utilizando protocolos estándar de comunicación y el cual permite configurar los parámetros de la captura de imágenes. En la tesis no se dio detalle del prototipo de captura de imágenes digitales, pues se considera material confidencial, debido a que será sometido a patente.

La etapa de pre procesado permite mejorar las características de la imagen que se ingresa a la etapa de OCR, buscando disminuir la tasa de error producida en la detección del texto. En esta etapa se desarrollaron cuatro métodos: el método TBH, el método KMN, el método MrRecon y la fusión de los métodos.

El método TBH se desarrolló utilizando como base de funcionamiento las operaciones morfológicas de Top Hat y Black Hat, las cuales permiten extraer regiones pequeñas; la extracción es definida por el elemento estructural con el que se realiza la operación. En esta tesis se propuso un elemento estructural en forma de una matriz de ceros de tamaño 81x81, con el renglón intermedio compuesto por unos, el cual permitió utilizar las operaciones morfológicas de Top Hat y Black Hat para segmentar correctamente el texto de la imagen. Este método funciona bien con aquellos cables en donde el texto tiene un contorno bien definido, y en donde el color del texto y el cable es contrastante. TBH resulta robusto en los casos donde el texto está tallado ya que el E. S. propuesto

reconstruye algunos de huecos del texto. TBH presenta desventaja en aquellos casos en donde el texto es borroso y la iluminación del cable es deficiente. En este último caso, se presentan sombras o brillos en el extremo cable, por lo cual se generan contornos en esta área del cable debido a que se puede considerar al cable como un elemento cilíndrico.

En el método KMN se empleó el método K-means para agrupar el texto, el cable y el fondo en tres distintos grupos, pero, como se discutió en la Sección 5.3.3, fue necesario considerar un cluster extra, el cual pudiese clasificar de manera correcta el borde del texto o los componentes correspondientes al ruido, y así permitir segmentar el texto de la imagen original. El método KMN funciona bien en aquellos cables en donde la iluminación genera brillo o sombras en el cable, y en donde el color del texto y el cable es contrastante, tolerando inclusive aquellos cables en donde el borde puede estar un poco borroso. Sin embargo, KMN presenta desventaja cuando el texto es moderadamente borroso o se presentan tallones en el cable.

En el método MrRecon se desarrolló utilizando como base el funcionamiento el método de reconstrucción morfológica por dilatación en donde los valores de alta intensidad remplazan a los valores vecinos de baja intensidad. Se definió un elemento estructural para determinar hasta qué punto se puede extender un valor en una imagen semilla acotada por una imagen máscara que especifica el valor máximo permitido en cada píxel. La imagen máscara limita la propagación de valores de alta intensidad, lo cual reconstruye todos los valores menores al límite establecido por la máscara; el resto de la imagen es reconstruido a partir de la resta de la imagen reconstruida a la imagen original, permitiendo de esta manera segmentar el texto de la imagen. MrRecon funciona bien en aquellos productos en donde el contraste del cable y el texto es alto o medio y en cables donde el borde del texto llega a ser borroso. Sin embargo, MrRecon presenta desventaja en aquellos productos donde el cable es moderadamente borroso o el cable presenta segmentos faltantes en el texto.

La fusión de métodos se desarrolló considerando la cuestión de que cada método tiene ventajas y desventajas; y considerando que uno de ellos puede validar algunos caracteres que en algún otro método pueden no validarse. Con en base en esto se diseñó un método que fusiona la validación de los métodos en una sola validación,

otorgándole robustez al sistema y disminuyendo la posibilidad de obtener falsos positivos o negativos. Al presentarse una repetibilidad de falla en la validación resultante de los métodos, es posible poder detectar problemas en el etiquetado. La fusión de métodos trabaja bien con todos los cables, ya que las ventajas de unos métodos compensan las desventajas de otros. Sin embargo, la fusión de métodos presenta desventaja en aquellos cables en donde el texto llega a ser muy borroso o el texto presenta una alta segmentación o fragmentación.

Las imágenes generadas por cada uno de los métodos pueden contener elementos correspondientes a ruido o defectos en el texto, por lo tanto, se diseñó el algoritmo S.A.F.E. el cual utiliza las características de los contornos en cada imagen; con base en estas características se creó un modelo de clasificación de contornos basado en reglas de decisión con la ayuda del software WEKA. El algoritmo JRIP fue el que permitió construir este modelo; estas reglas permitieron clasificar los contornos de la imagen ingresada al algoritmo S.A.F.E. para poder detectar y obtener la posición de los componentes correspondientes al ruido o los segmentos del texto en los cuales se presentan defectos. El algoritmo JRIP permitió obtener una precisión del 98.8% utilizando cinco reglas; la evaluación del algoritmo se realizó sobre un dataset formado por 2023 instancias, cada una con 7 atributos de cada uno de los contornos en la imagen, los contornos fueron clasificados en 3 clases, la primera, correspondiente a contornos de letras; la segunda, correspondiente a contornos de símbolos gráficos; y la tercera, correspondiente a contornos de ruido o manchas de impresión.

Para el algoritmo de OCR se utilizó el motor de Tesseract; este motor permite convertir una imagen con texto a un conjunto de datos, en el que se obtiene el número de palabras, las coordenadas de las palabras, la confianza de la predicción de la palabra y la predicción misma representada como una cadena de caracteres. A lo largo del desarrollo de este trabajo fue posible conocer algunas de las limitantes o características que permiten mejorar la predicción obtenida por el motor OCR. La etapa de pre procesamiento busca mejorar estos resultados; asimismo, el tamaño del texto, el contraste del texto con el fondo de la imagen y los bordes del texto representan parámetros que pueden mejorar o afectar el resultado del OCR.

De manera global la validación del resultado obtenido por la etapa de OCR, originalmente se realizó solamente con la distancia de Levenshtein y la distancia de Hamming; pero estas distancias permitían solamente conocer la cantidad de caracteres erróneos, por lo cual utilizar solamente esta métrica para validar el texto resultó en una validación parcial de la información. En el caso de estudio presentado en esta tesis, la etiqueta que es colocada en el conductor eléctrico está segmentada en un grupo de imágenes; por lo tanto, bajo este contexto y buscando complementar las métricas de distancia, se diseñó el algoritmo VALET, el cual permite validar el texto de la imagen contra el texto de la etiqueta; esta información siempre será conocida. VALET permite conocer cuáles caracteres son los que no se pudieron validar a lo largo de la secuencia de imágenes y evitar falsos positivos, inclusive si el texto empieza a ser reconocido o no corresponde al inicio de la etiqueta, ya que los requisitos para que el texto sea validado contra el del GT son dos: la posición y la equivalencia igual del carácter a validar. La validación del texto utilizando VALET permitió obtener una tasa de error promedio del 26.53% para el método TBH, 19.45% para el método KMN, 14.75% para el método MrRecon y 8.69% para la fusión de los métodos.

Finalmente, la última etapa del sistema es la de visualización de resultados. Esta etapa representa de manera visual la información obtenida de la etapa de OCR y del algoritmo S.A.F.E., representando de manera visual los resultados; además permite conocer al operador de manera sencilla el estado actual del sistema y en caso de presentarse errores o fallas conocer la sección del texto en donde se están presentando. En la interfaz propuesta se presenta la certeza de cada una de las palabras, un rectángulo el cual contiene a las palabras, los valores de las métricas de Levenshtein y Hamming, y un rectángulo conteniendo las áreas donde se han encontrado defectos de impresión. Para simplificar la visualización de resultados se creó una relación de colores: verde claro, para representar texto correcto considerando espacios y palabras; verde oscuro, para representar texto correcto utilizando un comodín; naranja, para representar texto correcto considerando solamente los caracteres que lo forman; y Rojo, para representar texto incorrecto.

El desarrollo del sistema de visión inteligente para detección de etiquetas en conductores eléctricos recubiertos propuesto buscó otorgar al proceso de extrusión

información que permita controlar el nivel de calidad del producto o detectar problemas durante el proceso de fabricación. La complejidad del problema estudiado y los resultados obtenidos y presentados en la Sección 5.4 y 5.5 demuestran que el diseño de este sistema es factible para su implementación en campo.

## APORTACIÓN DE LA TESIS

Las aportaciones de la tesis son:

- Método de segmentación de texto basado en operaciones morfológicas y creación de un elemento estructural diseñado para el caso de estudio, denominado TBH Engine (presentado en la Sección 4.3.1).
- Método de segmentación de texto basado en clusterización a través de K-medias, denominado KMN Engine (presentado en la Sección 4.3.2).
- Método de segmentación de texto basado en reconstrucción morfológica por dilatación, denominado MrRecon Engine (presentado en la Sección 4.3.3).
- Método de fusión utilizando la validación de cada uno de los métodos individuales: TBH Engine, KMN Engine y MrRecon Engine (presentado en la Sección 4.3.4).
- Algoritmo para la detección de ruido y defectos en el texto utilizando un algoritmo de clasificación basado en reglas, denominado S.A.F.E. (presentado en la Sección 4.4).
- Algoritmo de validación del texto utilizando secuencias de imágenes y basado en un acumulador de aciertos o verdaderos positivos (TP), denominado VALET (presentado en la Sección 4.6.3).
- Método para visualizar el resultado del sistema, que de manera gráfica permita resaltar los errores o defectos en la impresión y marcar el estatus de la detección del texto de la etiqueta utilizando un código de colores (presentado en la Sección 4.7).

## **APORTACIÓN SOCIAL DE LA TESIS**

Este trabajo de tesis beneficia al sector de la industria de los conductores eléctricos y al sector de la manufactura avanzada a través de las siguientes aportaciones:

- La disminución de plástico y tinta a desechar por la detección temprana de defectos o errores en el etiquetado del cable.
- La mejora en la calidad del producto manufacturado, ya que el sistema permite tener un control más estricto en la calidad del etiquetado.
- La implementación de sistemas de visión en el sector de fabricación de conductores eléctricos.
- La disminución de la fatiga visual del operador, ya que deja de ser necesario utilizar una lámpara estroboscópica para la validación manual del etiquetado.
- El conocimiento generado con la tesis se compartirá en un repositorio público a través de la publicación de la tesis.

## RECOMENDACIONES

Los sistemas de visión aplicados a la industria de los conductores eléctricos son escasos. Por lo tanto, diseñar e implementar nuevas soluciones o la integración de tecnologías actuales para solucionar las distintas problemáticas en este sector industrial se consideran un campo de desarrollo interesante.

Como trabajo futuro se proponen los siguientes puntos:

1. Paralelizar los métodos desarrollados para poder disminuir el tiempo de ejecución de cada uno y poder aplicarlos en líneas de alta velocidad.
2. Diseñar una etapa de pre procesamiento y OCR utilizando técnicas de aprendizaje profundo diseñadas para el caso en específico de la detección de texto en conductores eléctricos recubiertos.
3. Diseñar una Interfaz gráfica que permita mostrar información detallada sobre el proceso, como: el tipo de falla, un reporte de fallas, un explorador o visualizador de las imágenes con falla.
4. Diseñar un sistema que permita validar el color del cable; para ciertos fabricantes, el color del cable está regulado por estándares de calidad, incluso en algunos procesos de extrusión se coloca una franja de otro color distinto a la del cable.

## REFERENCIAS BIBLIOGRÁFICAS

- [1] Giró, S, "La visión artificial en la industria", *INFAIMON*, 2016. <https://blog.infaimon.com/la-vision-artificial-industria-4-0/> (consultado oct. 20, 2020).
- [2] D. Marr, *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. Henry Holt and Co., Inc., 1982.
- [3] Fernández García N.L., "Introducción a la visión artificial", Universidad de Córdoba, 2012, [En línea]. Disponible en: <http://www.uco.es/users/ma1fegan/2011-2012/vision/Temas/Vision-artificial.pdf>.
- [4] "Cable Extrusion", *SAMPSISTEMI*. <https://www.sampsistemi.com/cable-extrusion/> (consultado jul. 19, 2020).
- [5] "Wire and cable marking machines". <https://www.schleuniger.com/en-us/products/peripherals/markings/-printing/-labeling/> (consultado oct. 20, 2020).
- [6] "Coding & Marking Printers | Gem Gravure". <https://www.gemgravure.com/coding-marking-printers/> (consultado jul. 19, 2020).
- [7] "Impresoras para tubos y cables | Videojet México". <https://www.videojet.mx/mx/homepage/industry-solutions/wire-cable-pipe.html> (consultado ago. 12, 2019).
- [8] TopCable, "tipos de recubrimiento | Top Cable", *Cables y consejos eléctricos*. By Top Cable. <https://www.topcable.com/blog-electric-cable/tag/tipos-de-recubrimiento/> (consultado ago. 20, 2020).
- [9] G. F. Moore y BICC Cables Ltd, Eds., *Electric cables handbook*, 3rd ed. Oxford; Malden, Mass: Blackwell Science, 1997.
- [10] "Codificadores láser: grabado y marcaje con láser | Videojet México". <https://www.videojet.mx/mx/homepage/products/laser-marking-systems.html> (consultado jul. 04, 2019).
- [11] "METZNER | Thermal transfer printer for printing on cables". <https://www.metzner.com/en/products/cable-processing-corrugated-tubes/peripherals-cable/printing-marking/metzner-thermal-transfer-printers-cable.html> (consultado ago. 13, 2019).
- [12] National Fire Protection Association, "NFPA 70®: National Electrical Code®". NFPA, 2020, consultado: ago. 30, 2019. [En línea]. Disponible en: <https://www.nfpa.org/codes-and-standards/all-codes-and-standards/list-of-codes-and-standards/detail?code=70>.
- [13] "¿Qué tipos de cables eléctricos existen?" <https://masvoltaje.com/blog/tipos-de-cables-electricos-que-existen-n12> (consultado jul. 04, 2019).
- [14] B01 Committee, "Specification for Standard Nominal Diameters and Cross-Sectional Areas of AWG Sizes of Solid Round Wires Used as Electrical Conductors", ASTM International. doi: 10.1520/B0258-18.
- [15] "Ance normalización en línea". <http://www.ance.org.mx/NormalizacionOnLine/TiposNormas.aspx?IdTipoNorma=4> (consultado ago. 14, 2019).
- [16] G. G. L. Enrique Sucar, *Visión Computacional*. INAOE Puebla, México.
- [17] T. de Souza Alves, C. S. de Oliveira, C. Sanin, y E. Szczerbicki, "From Knowledge based Vision Systems to Cognitive Vision Systems: A Review", *Procedia Comput. Sci.*, vol. 126, pp. 1855–1864, ene. 2018, doi: 10.1016/j.procs.2018.08.077.

- [18] E. Gutierrez Alegre, M. Pajares, y A. de la Escalera Hueso, *Conceptos y métodos en visión por computador*. España: s.l., 2016.
- [19] K. N. Joshi y B. T. Patil, "Effect of Illumination Systems on Statistical Texture Parameters Based Clustering and Discrimination of Machined Surfaces Using Machine Vision", *MAPAN*, pp. 1–9, jul. 2018, doi: 10.1007/s12647-018-0279-z.
- [20] Y. Caulier, K. Spinnler, S. Bourennane, y T. Wittenberg, "New Structured Illumination Technique for the Inspection of High-Reflective Surfaces: Application for the Detection of Structural Defects without any Calibration Procedures", *EURASIP J. Image Video Process.*, vol. 2008, núm. 1, p. 237459, dic. 2007, doi: 10.1155/2008/237459.
- [21] Axis Communications, "CCD and CMOS sensor technology Technical white paper". Axis Communications, 2010, consultado: mar. 26, 2019. [En línea]. Disponible en: [https://www.axis.com/files/whitepaper/wp\\_ccd\\_cmos\\_40722\\_en\\_1010\\_lo.pdf](https://www.axis.com/files/whitepaper/wp_ccd_cmos_40722_en_1010_lo.pdf).
- [22] D. L. Gilblom, "Cameras", *Mach. Vis. Handb.*, pp. 355–476, 2012, doi: 10.1007/978-1-84996-169-1\_10.
- [23] Chia-Kai Liang, Li-Wen Chang, y H. H. Chen, "Analysis and Compensation of Rolling Shutter Effect", *IEEE Trans. Image Process.*, vol. 17, núm. 8, pp. 1323–1330, ago. 2008, doi: 10.1109/TIP.2008.925384.
- [24] G. Liu, Y. Sun, y X. Chen, "A Rectification Method for Removing Rolling Shutter Distortion", *Proc. 2012 Int. Conf. Inf. Technol. Softw. Eng.*, pp. 235–245, 2013, doi: 10.1007/978-3-642-34528-9\_25.
- [25] S. P. Nicklin, R. D. Fisher, y R. H. Middleton, "Rolling Shutter Image Compensation", en *RoboCup 2006: Robot Soccer World Cup X*, jun. 2006, pp. 402–409, doi: 10.1007/978-3-540-74024-7\_39.
- [26] R. Rodríguez y J. Carlos, "MONICOD: supervisión visual a muy alta velocidad de codificación impresa industrial", 2012, consultado: sep. 01, 2019. [En línea]. Disponible en: <https://accedacris.ulpgc.es/jspui/handle/10553/10475>.
- [27] B. AG, "Global Shutter, Rolling Shutter: White Paper Update". ene. 08, 2014, consultado: sep. 01, 2019. [En línea].
- [28] Andor Learning Centre, "Rolling Shutter vs Global Shutter Technical Note". oct. 10, 2012, consultado: sep. 01, 2019. [En línea]. Disponible en: <https://andor.oxinst.com/learning/view/article/rolling-and-global-shutter>.
- [29] A. A. Tazehkandi y A. Ahmadi Tazehkandi, *Hands-On Algorithms for Computer Vision*. S.l.: Packt Publishing, 2018.
- [30] W. Burger, *Digital image processing: an algorithmic introduction using Java*. New York, NY: Springer Berlin Heidelberg, 2016.
- [31] R. C. Gonzalez y R. E. Woods, *Digital image processing*, Fourth edition. New York, NY: Pearson, 2018.
- [32] J. Chaki y N. Dey, *A beginner's guide to image preprocessing techniques*. Boca Raton: Taylor & Francis, a CRC title, part of the Taylor & Francis imprint, a member of the Taylor & Francis Group, the academic division of T&F Informa, plc, 2019.
- [33] J. Beyerer, F. Puente León, y C. Frese, *Machine vision: automated visual inspection: theory, practice and applications*. Berlin Heidelberg: Springer, 2016.
- [34] S. Krig, *Computer vision metrics: survey, taxonomy, and analysis*. New York, NY: Apress, 2014.
- [35] G. Mehul, P. Ankita, D. Namrata, G. Rahul, y S. Sheth, "Text-based Image Segmentation Methodology", *Procedia Technol.*, vol. 14, pp. 465–472, ene. 2014, doi: 10.1016/j.protcy.2014.08.059.

- [36] J. B. Pedersen, K. Nasrollahi, y T. B. Moeslund, "Quality inspection of printed texts", en *2016 International Conference on Systems, Signals and Image Processing (IWSSIP)*, Bratislava, Slovakia, may 2016, pp. 1–4, doi: 10.1109/IWSSIP.2016.7502718.
- [37] P. Soille, *Morphological Image Analysis: Principles and Applications*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004.
- [38] D. S. Doermann y K. Tombre, *Handbook of document image processing and recognition*. 2014.
- [39] A. Chaudhuri, K. Mandaviya, P. Badelia, y S. K Ghosh, *Optical Character Recognition Systems for Different Languages with Soft Computing*, vol. 352. Cham: Springer International Publishing, 2017.
- [40] "tesseract-ocr/tesseract", *GitHub*. <https://github.com/tesseract-ocr/tesseract> (consultado sep. 21, 2019).
- [41] P. J. Schneider, "An algorithm for automatically fitting digitized curves", en *Graphics Gems*, Elsevier, 1990, pp. 612–626.
- [42] P. J. Rousseeuw y A. M. Leroy, *Robust regression and outlier detection*. New York: Wiley, 1987.
- [43] R. Smith, "An Overview of the Tesseract OCR Engine", en *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007) Vol 2*, Curitiba, Parana, Brazil, sep. 2007, pp. 629–633, doi: 10.1109/ICDAR.2007.4376991.
- [44] R. W. Smith, "The extraction and recognition of text from multimedia document images", University of Bristol, 1987.
- [45] S. Hochreiter y J. Schmidhuber, "Long Short-Term Memory", *Neural Comput.*, vol. 9, núm. 8, pp. 1735–1780, nov. 1997, doi: 10.1162/neco.1997.9.8.1735.
- [46] S. Raschka y V. Mirjalili, *Python machine learning: machine learning and deep learning with Python, scikit-learn, and TensorFlow*, Second edition, Fourth release, [fully revised and Updated]. Birmingham Mumbai: Packt Publishing, 04.
- [47] I. H. Witten y E. Frank, *Data mining: practical machine learning tools and techniques*, 2nd ed. Amsterdam ; Boston, MA: Morgan Kaufman, 2005.
- [48] K. M. Ting, "Precision and Recall", en *Encyclopedia of Machine Learning*, C. Sammut y G. I. Webb, Eds. Boston, MA: Springer US, 2010, pp. 781–781.
- [49] P. M. Manwatkar y K. R. Singh, "A technical review on text recognition from images", en *2015 IEEE 9th International Conference on Intelligent Systems and Control (ISCO)*, Coimbatore, India, ene. 2015, pp. 1–5, doi: 10.1109/ISCO.2015.7282362.
- [50] D. Chen, J.-M. Odobez, y H. Bourlard, "Text detection and recognition in images and video frames", *Pattern Recognit.*, vol. 37, núm. 3, pp. 595–608, mar. 2004, doi: 10.1016/j.patcog.2003.06.001.
- [51] A. Choudhary, R. Rishi, y S. Ahlawat, "A New Character Segmentation Approach for Off-Line Cursive Handwritten Words", *Procedia Comput. Sci.*, vol. 17, pp. 88–95, ene. 2013, doi: 10.1016/j.procs.2013.05.013.
- [52] J. C. Rodríguez-Rodríguez, A. Quesada-Arencibia, R. Moreno-Díaz, y C. R. García, "A Character Segmentation Proposal for High-Speed Visual Monitoring of Expiration Codes on Beverage Cans", *Sensors*, vol. 16, núm. 4, abr. 2016, doi: 10.3390/s16040527.
- [53] S. N. Hashmi, K. Kumar, S. Khandelwal, D. Lochan, y S. Mittal, "Real Time License Plate Recognition from Video Streams using Deep Learning", *Int. J. Inf. Retr. Res.*, vol. 9, núm. 1, pp. 65–87, mar. 2019, doi: 10.4018/IJIRR.2019010105.
- [54] A. Farhat *et al.*, "Optical character recognition on heterogeneous SoC for HD automatic number plate recognition system", *EURASIP J. Image Video Process.*, vol. 2018, núm. 1, p. 58, dic. 2018, doi: 10.1186/s13640-018-0298-2.

- [55] Hendry y R.-C. Chen, "Automatic License Plate Recognition via sliding-window darknet-YOLO deep learning", *Image Vis. Comput.*, vol. 87, pp. 47–56, jul. 2019, doi: 10.1016/j.imavis.2019.04.007.
- [56] A. Fischer, V. Frinken, y H. Bunke, "Chapter 17 - Hidden Markov Models for Off-Line Cursive Handwriting Recognition", en *Handbook of Statistics*, vol. 31, C. R. Rao y V. Govindaraju, Eds. Elsevier, 2013, pp. 421–442.
- [57] G. Hsu, J. Chen, y Y. Chung, "Application-Oriented License Plate Recognition", *IEEE Trans. Veh. Technol.*, vol. 62, núm. 2, pp. 552–561, feb. 2013, doi: 10.1109/TVT.2012.2226218.
- [58] F. De Sousa Ribeiro *et al.*, "An End-to-End Deep Neural Architecture for Optical Character Verification and Recognition in Retail Food Packaging", en *2018 25th IEEE International Conference on Image Processing (ICIP)*, Athens, oct. 2018, pp. 2376–2380, doi: 10.1109/ICIP.2018.8451555.
- [59] E. Shelhamer, J. Long, y T. Darrell, "Fully Convolutional Networks for Semantic Segmentation", *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, núm. 4, pp. 640–651, abr. 2017, doi: 10.1109/TPAMI.2016.2572683.
- [60] A. K. Bhunia, G. Kumar, P. P. Roy, R. Balasubramanian, y U. Pal, "Text recognition in scene image and video frame using Color Channel selection", *Multimed. Tools Appl.*, vol. 77, núm. 7, pp. 8551–8578, abr. 2018, doi: 10.1007/s11042-017-4750-6.
- [61] S. M. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong, y R. Young, "ICDAR 2003 robust reading competitions", en *Seventh International Conference on Document Analysis and Recognition, 2003. Proceedings.*, ago. 2003, pp. 682–687, doi: 10.1109/ICDAR.2003.1227749.
- [62] G. Zhong y M. Cheriet, "Tensor representation learning based image patch analysis for text identification and recognition", *Pattern Recognit.*, vol. 48, núm. 4, pp. 1211–1224, abr. 2015, doi: 10.1016/j.patcog.2014.09.025.
- [63] N. V. Rao, "OPTICAL CHARACTER RECOGNITION TECHNIQUE ALGORITHMS", *Journal of Theoretical and Applied Information Technology*, vol. 83, no. 2, p. 8, 2015.
- [64] B. Y. Amirgaliyev, K. K. Kuvatov, y Z. Y. Baibatyr, "Application of Convolutional Neural Network for Optical Character Recognition Designed for Kazakhstan Identity Cards", en *2017 IEEE 11th International Conference on Application of Information and Communication Technologies (AICT)*, Moscow, Russia, sep. 2017, pp. 1–3, doi: 10.1109/ICAICT.2017.8686945.
- [65] T. C. Wei, U. U. Sheikh, y A. A.-H. A. Rahman, "Improved optical character recognition with deep neural network", en *2018 IEEE 14th International Colloquium on Signal Processing & Its Applications (CSPA)*, Batu Feringghi, mar. 2018, pp. 245–249, doi: 10.1109/CSPA.2018.8368720.
- [66] "About the free OCR API". <http://ocr.space/about> (consultado sep. 22, 2019).
- [67] H. Modi y M. C., "A Review on Optical Character Recognition Techniques", *Int. J. Comput. Appl.*, vol. 160, núm. 6, pp. 20–24, feb. 2017, doi: 10.5120/ijca2017913061.
- [68] L. von Ahn, M. Blum, N. J. Hopper, y J. Langford, "CAPTCHA: Using Hard AI Problems for Security", en *Advances in Cryptology — EUROCRYPT 2003*, 2003, pp. 294–311.
- [69] M. A. H. Monil, M. S. Q. Z. Nine, B. Poon, M. A. Amini, y H. Yan, "Bangla text processing and recognition based on Fuzzy unsupervised Feature Extraction and SVM", en *2013 International Conference on Machine Learning and Cybernetics*, jul. 2013, vol. 03, pp. 1272–1278, doi: 10.1109/ICMLC.2013.6890784.
- [70] T. Hassan y H. A. Khan, "Handwritten Bangla numeral recognition using Local Binary Pattern", en *2015 International Conference on Electrical Engineering and*

- Information Communication Technology (ICEEICT)*, may 2015, pp. 1–4, doi: 10.1109/ICEEICT.2015.7307371.
- [71] C. Liyanage, T. Nadungodage, y R. Weerasinghe, “Developing a commercial grade Tamil OCR for recognizing font and size independent text”, en *2015 Fifteenth International Conference on Advances in ICT for Emerging Regions (ICTer)*, ago. 2015, pp. 130–134, doi: 10.1109/ICTER.2015.7377678.
- [72] A. Farhat *et al.*, “OCR based feature extraction and template matching algorithms for Qatari number plate”, en *2016 International Conference on Industrial Informatics and Computer Systems (CIICS)*, mar. 2016, pp. 1–5, doi: 10.1109/ICCSII.2016.7462419.
- [73] S. Pasha y M. C. Padma, “Handwritten Kannada character recognition using wavelet transform and structural features”, en *2015 International Conference on Emerging Research in Electronics, Computer Science and Technology (ICERECT)*, dic. 2015, pp. 346–351, doi: 10.1109/ERECT.2015.7499039.
- [74] R. A. Bedruz, E. Sybingco, A. R. Quiros, A. C. Uy, R. R. Vicerra, y E. Dadios, “Fuzzy logic based vehicular plate character recognition system using image segmentation and scale-invariant feature transform”, en *2016 IEEE Region 10 Conference (TENCON)*, nov. 2016, pp. 676–681, doi: 10.1109/TENCON.2016.7848088.
- [75] W. Q. Khan y R. Q. Khan, “Urdu optical character recognition technique using point feature matching; a generic approach”, en *2015 International Conference on Information and Communication Technologies (ICICT)*, dic. 2015, pp. 1–7, doi: 10.1109/ICICT.2015.7469576.
- [76] H. Lin y C. Hsu, “Optical character recognition with fast training neural network”, en *2016 IEEE International Conference on Industrial Technology (ICIT)*, mar. 2016, pp. 1458–1461, doi: 10.1109/ICIT.2016.7474973.
- [77] A. Singh y S. Desai, “Optical character recognition using template matching and back propagation algorithm”, en *2016 International Conference on Inventive Computation Technologies (ICICT)*, ago. 2016, vol. 3, pp. 1–6, doi: 10.1109/INVENTIVE.2016.7830161.
- [78] G. Van Rossum y F. L. Drake, *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace, 2009.
- [79] G. Bradski, “The OpenCV Library”, *Dr Dobbs J. Softw. Tools*, 2000.
- [80] “PyCharm: el IDE de Python para desarrolladores profesionales, por JetBrains”, *JetBrains*. <https://www.jetbrains.com/es-es/pycharm/> (consultado mar. 31, 2020).
- [81] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, y I. H. Witten, “The WEKA data mining software: an update”, *ACM SIGKDD Explor. Newsl.*, vol. 11, núm. 1, pp. 10–18, nov. 2009, doi: 10.1145/1656274.1656278.
- [82] I. H. Witten y I. H. Witten, Eds., *Data mining: practical machine learning tools and techniques*, Fourth Edition. Amsterdam: Elsevier, 2017.
- [83] S. van der Walt *et al.*, “scikit-image: image processing in Python”, *PeerJ*, vol. 2, p. e453, 2014, doi: 10.7717/peerj.453.
- [84] “OpenCV: Color conversions”. [https://docs.opencv.org/3.4/de/d25/imgproc\\_color\\_conversions.html#color\\_convert\\_rgb\\_lab](https://docs.opencv.org/3.4/de/d25/imgproc_color_conversions.html#color_convert_rgb_lab) (consultado ago. 12, 2020).
- [85] A. Kaehler y G. R. Bradski, *Learning OpenCV 3: computer vision in C++ with the OpenCV library*, First edition, Second release. Sebastopol, CA: O'Reilly Media, 2017.
- [86] N. Otsu, “A Threshold Selection Method from Gray-Level Histograms”, *IEEE Trans. Syst. Man Cybern.*, vol. 9, núm. 1, pp. 62–66, ene. 1979, doi: 10.1109/TSMC.1979.4310076.

- [87] J. Howse, P. Joshi, y M. Beyeler, *OpenCV: computer vision projects with Python: get savvy with OpenCV and actualize cool computer vision applications: a course in three modules*. Birmingham Mumbai: Packt, 2016.
- [88] A. Fernández Villán, *MASTERING OPENCV 4 WITH PYTHON: a practical guide covering topics from image processing, ... augmented reality to deep learning with opencv 4 a*. Place of publication not identified: PACKT Publishing Limited, 2019.
- [89] K. Robinson y P. F. Whelan, "Efficient morphological reconstruction: a downhill filter", *Pattern Recognit. Lett.*, vol. 25, núm. 15, pp. 1759–1767, nov. 2004, doi: 10.1016/j.patrec.2004.07.002.
- [90] "OpenCV: Contour Features". [https://docs.opencv.org/4.4.0/dd/d49/tutorial\\_py\\_contour\\_features.html](https://docs.opencv.org/4.4.0/dd/d49/tutorial_py_contour_features.html) (consultado jul. 19, 2020).
- [91] "OpenCV: Contour Properties". [https://docs.opencv.org/4.4.0/d1/d32/tutorial\\_py\\_contour\\_properties.html](https://docs.opencv.org/4.4.0/d1/d32/tutorial_py_contour_properties.html) (consultado jul. 19, 2020).
- [92] W. W. Cohen, "Fast Effective Rule Induction", en *Twelfth International Conference on Machine Learning*, 1995, pp. 115–123.
- [93] D. S. Doermann, Ed., *Handbook of document image processing and recognition: with 98 tables*. London: Springer Reference, 2014.
- [94] G. Navarro, "A guided tour to approximate string matching", *ACM Comput. Surv.*, vol. 33, núm. 1, pp. 31–88, mar. 2001, doi: 10.1145/375360.375365.