

ARTICLE HISTORY

Received 11 September 2020
Accepted 16 October 2020

Juan J. Navarro

Department of Automation projects
Continuous Improvement Management
Associates S.C.
San Luis Potosí, México
jnavarro@cima.us
<https://orcid.org/0000-0002-5674-0265>

Carolina Reta

Department of IT, Control, and Electronics
CONACYT-CIATEQ A. C.
Hidalgo, México.
carolina.reta@ciateq.mx
<https://orcid.org/0000-0002-0843-129X>

Procesamiento digital de imágenes en la creación de prototipo de sistema inteligente para detección y reconocimiento de texto en etiquetas de conductores eléctricos recubiertos

*Digital image processing in
the creation of an intelligent
system prototype for text
detection and recognition
in the labeling process of
electrical cable*

Procesamiento digital de imágenes en la creación de prototipo de sistema inteligente para detección y reconocimiento de texto en etiquetas de conductores eléctricos recubiertos

Digital image processing in the creation of an intelligent system prototype for text detection and recognition in the labeling process of electrical cable

Juan J. Navarro

Master's program
CIATEQ A. C.
Department of Automation projects
Continuous Improvement Management
Associates S.C.
San Luis Potosi, Mexico
jnavarro@cima.us

Carolina Reta

Department of IT, Control, and Electronics
CONACYT-CIATEQ A. C.
Hidalgo, México.
carolina.reta@ciateq.mx

Abstract — El etiquetado del cable permite identificar distintas configuraciones y lotes de cables, así como conocer sus características. En el proceso de etiquetado del cable se pueden presentar distintos tipos de errores o defectos en el texto impreso, por ejemplo: ausencia de la etiqueta, corrimiento de la tinta, partes del texto faltante, texto ilegible y gotas de tinta. En este trabajo se presenta un prototipo de sistema para la validación del texto en cables eléctricos mediante técnicas de procesamiento de imágenes. El sistema propuesto integra dos etapas. En la primera etapa se propuso un método basado en clusterización por K-means que permite realizar el preprocesado de las imágenes para acondicionarlas. En la segunda etapa se realiza el reconocimiento óptico de caracteres utilizando el motor Tesseract OCR, el cual permite convertir el texto contenido en las imágenes en cadenas de caracteres. La experimentación se realizó utilizando una colección de 909 imágenes de cables eléctricos que contiene ocho tipos diferentes de cables, donde las imágenes están etiquetadas individualmente con la ground truth. La experimentación realizada permitió obtener una tasa de error promedio del 6.54%, 3.97% y 2.53% al validar el texto empleando tres, cinco y siete secuencias de etiquetas, respectivamente.

Keywords — Detección de texto en cables; sistemas inteligentes; procesamiento digital de imágenes; K-means; OCR; Tesseract.

Abstract — Cable labeling allows identifying different configurations and batches of cables, as well as their characteristics. In the cable labeling process, different types of errors or defects can occur in the printed text, such as the absence of the label, ink bleeding, text parts missing, illegible text, and ink drops. In this paper, a prototype of a system for text validation in electrical cables using image processing techniques is presented. The proposed system consists of two stages. In the first stage, a method based on K-means clustering was proposed, allowing preprocessing images to condition them. In the second stage, optical character recognition using the Tesseract OCR engine is performed, allowing the text in the images to be converted into character strings. The experimentation was carried out using a collection of 909 images of electrical cables containing eight different types of cables, where the images are individually labeled with the ground truth. The evaluation obtained an average error rate of 6.54%, 3.97%, and 2.53% when validating the text using three, five, and seven label sequences, respectively.

Keywords — Text detection on cables; intelligent systems; digital image processing; K-means; OCR; Tesseract

I. INTRODUCCIÓN

En el sector industrial de la fabricación de conductores eléctricos poco a poco se van implementando herramientas tecnológicas para lograr su conversión en una Industria 4.0. Dentro de los diferentes procesos que se realizan para la fabricación de cable eléctrico existe el proceso de extrusión[1], en el cual se aplica 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 [2]-[4]. 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.

La problemática principal relacionada con el etiquetado 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. Otra problemática 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, alterando la planificación de producción, 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 los problemas de ausencia e impresión con información errónea de las etiquetas, permitiendo a su vez cumplir con estándares de calidad para el producto.

Por lo anteriormente expuesto, desarrollar un sistema inteligente basado en técnicas de procesamiento de imágenes que sea 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.

II. ESTADO DEL ARTE

A continuación, se presenta el trabajo relacionado de sistemas OCR (del inglés Optical Character Recognition) aplicados tanto en el reconocimiento de texto en documentos como en sistemas inteligentes de inspección visual de productos; también se presentan distintas técnicas para optimizar la etapa de preprocesado de imágenes y distintos métodos para clasificar caracteres utilizando enfoques de aprendizaje automático.

A. Reconocimiento de texto en documentos

El reconocimiento de texto en los documentos presenta diversas problemáticas que deben abordarse considerando: el área de estudio y tipo de documento con el que se trabaja; el origen; la manera de digitalizarlo; y el tipo de texto que contiene, ya sea impreso o

escrito a mano. Dos trabajos destacados que abordan estas problemáticas se presentan a continuación. En [5], utilizando una SVM se procesa el canal que permite obtener mejores características de la imagen de manera automática, combinándolo con un modelo de redes ocultas de Markov para el reconocimiento de palabras. Se obtuvo una precisión del 75.41% en el conjunto ICDAR 2013 [6] y 72.78% en el conjunto de textos propio Devanagari. En [7] se presenta un algoritmo de reconocimiento de caracteres escritos a mano basado en redes neuronales, el cual logra una tasa de reconocimiento de 95.44% utilizando un dataset propio para la evaluación del sistema. Las distintas técnicas utilizadas para el reconocimiento de texto en documentos no son completamente viables para el presente trabajo, debido a que, en la mayoría de estos

casos, el color del texto es contrastante con el color del fondo, situación que no siempre se presenta en el caso de estudio, ya que existen distintos colores de cables, así como colores de etiquetas. En el presente trabajo, para otorgar robustez al reconocimiento de texto en documentos, se utilizó el motor de OCR Tesseract versión 4.0, el cual contiene un algoritmo de reconocimiento basado en redes neuronales.

Los sistemas de OCR pueden aplicarse en: "recopilación de datos bancarios, facturas, documentos legales, incluso en el reconocimiento de números y escritura a mano; también pueden usarse para control de inventario en repositorios institucionales y bibliotecas digitales, reconocimiento óptico de música" [7] y análisis de vulnerabilidades en algoritmos CAPTCHA [8].

B. Sistemas inteligentes de inspección de productos

La inspección visual de productos utilizando sistemas inteligentes, requiere de un desarrollo que sea robusto a las distintas dificultades que se pueden presentar y que pueda trabajar en tiempo real. A continuación, se presentan dos ejemplos de sistemas de inspección basados en reconocimiento de texto.

En [9] 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, basado en el cálculo de 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, utilizando un dataset propio para la aplicación. En [10], 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 que incluye 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) [11] para la localización de la fecha. Este sistema obtuvo una precisión del 98% utilizando un dataset propio para la aplicación. En el presente trabajo, el prototipo de sistema inteligente de visión está dirigido a la detección y reconocimiento de etiquetas en conductores eléctricos recubiertos, 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 conlleva este análisis se encuentran: la variación entre el color del texto y el cable; y el tamaño del texto que varía dependiendo del tamaño del conductor eléctrico y del fabricante.

C. Optimización del pre procesado de la imagen capturada

Optimizar la etapa de pre procesado permite mejorar las características de la imagen de entrada al sistema OCR. A continuación, se presenta trabajo relacionado relevante en este tópico.

En [12] se 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 de reconocimiento de texto. El cual se divide 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; para este análisis se utilizó un dataset de las olimpiadas de 1992 en Barcelona, provisto por la BBC[13]. En [14] 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. En [15] 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, utilizando un dataset propio para la aplicación. En [16] 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 es del 99.4%; en este trabajo se utilizó un dataset propio para la evaluación del sistema. En [17] se realiza reducción de ruido, filtrados Gaussianos pasa bajos, corrección de inclinación con KSC, normalización del tamaño de la imagen, extracción de características con patrones locales binarios y clasificación utilizando el vecino euclidiano más cercano. Se logró obtener en el sistema una precisión del 96.5% al 96.7%, dependiendo del caso, utilizando la base de datos numérica manuscrita de Bangia CMATERdb [18]. En [19] 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, utilizando un dataset propio para el experimento. En [20] se utiliza el motor Tesseract, el cual contiene una función 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,

utilizando un dataset propio para el experimento. En [21] se utiliza template matching para extraer las características de la imagen; las cuales se comparan con una base de datos para elegir el carácter con el valor más alto de coincidencia empleando un algoritmo de back propagation. El algoritmo presenta problemas 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, utilizando un dataset propio para el experimento.

Combinando distintas técnicas de pre procesado, segmentación y clasificación, es posible lograr maximizar los resultados obtenidos de un algoritmo de OCR. En el sistema presentado en este artículo se propone una técnica para mejorar el pre procesado y la extracción de ruido en las imágenes digitales; posteriormente, se utiliza el motor de OCR Tesseract para convertir la imagen a texto; finalmente, se realiza la validación del texto para interpretar si el resultado contiene problemas o errores en el etiquetado.

En la Tabla I se presenta una comparación de las características de algunos trabajos con mayor similitud al propuesto, comparando: espacios de color, la robustez a diferentes situaciones que causan problemas, distintas técnicas para segmentación y métodos de clasificación del texto.

TABLA I. TRABAJO RELACIONADO VERSUS TRABAJO PROPUESTO

Ref.	Espacio de color				Robusto a				Pre procesamiento	Segmentación						Método de clasificación						
	RGB	LAB	Grises	Blanco y Negro	Escala Interlineado	Objetos en movimiento	Ruido en la imagen	Imágenes de baja calidad		Operación Morfológicas	Contornos	Tesseract	SVM	Umbralizado	DNN	K-means	LBP	DNN	HMM	NN	SVM	Tesseract
[7]			X		X	X	X	X	X		X						X					
[5]	X	X	X		X	X		X	X			X					X					
[9]			X				X	X	X	X			X						X			
[10]			X		X			X	X					X			X					
[16]	X			X	X				X	X		X									X	
[17]				X	X				X							X	X				X	
[20]				X	X			X						X								X
Propuesto	X				X	X	X		X				X		X							X

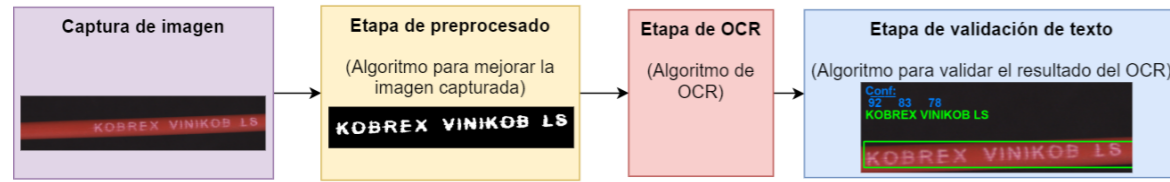


Fig. 1 Esquema del procesamiento digital de imágenes del prototipo de sistema inteligente.

III. MÉTODO PROPUESTO

El sistema propuesto está enfocado en la detección del texto impreso en conductores eléctricos; dicho texto está formado en su totalidad por tomas continuas de imágenes del cable estudiado. Por tanto, es necesario procesar cada imagen capturada en su momento y acumular el resultado obtenido. Dentro de los problemas que conlleva el análisis de las imágenes para la detección del texto se encuentran: la variación entre el color del texto y el cable; y el tamaño del texto que varía dependiendo del tamaño del conductor eléctrico y del fabricante. Para abordar estas problemáticas, en el sistema propuesto se segmenta el texto de la imagen utilizando el algoritmo de clusterización de K-means para posteriormente realizar el reconocimiento del mismo. En el sistema propuesto se adopta, por su facilidad de implementación y robustez, la versión 4.0 de Tesseract [22] junto con los paquetes de idiomas de español e inglés; al utilizar esta versión de Tesseract, que utiliza un motor basado en redes neuronales y LSTM, se espera obtener un mejor resultado comparado con el motor de OCR de las versiones anteriores.

El prototipo de sistema inteligente se divide en dos etapas: pre procesamiento y OCR. La Figura 1 presenta los bloques que componen el esquema general del procesamiento y análisis de las imágenes para la inspección de etiquetas en conductores eléctricos. El prototipo permite realizar la validación del texto que se imprime en el conductor eléctrico que se está fabricando mediante una inspección visual. El sistema prototipo funciona de la siguiente manera: 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; posteriormente se utiliza el algoritmo de OCR para convertir la imagen a texto; finalmente, se analiza el texto obtenido del algoritmo de OCR y se valida si el resultado es correcto o erróneo.

La implementación y desarrollo de los algoritmos propuestos en este sistema se

realizó utilizando software de fuente abierto: Python [23] y librerías de OpenCV [24].

A. Etapa de pre procesamiento

La etapa de pre procesamiento permite mejorar las características de la imagen de entrada. Para esta etapa se desarrolló un algoritmo basado en el método de clusterización de K-means que integra los siguientes pasos:

1) Cambio de espacio de color CIE L*a*b*

La imagen de entrada IMG se encuentra en el espacio de color RGB por lo que se realiza el cambio al espacio de color CIE L*a*b*. Este espacio está definido por el conjunto de Ecuaciones 1 [25].

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

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

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

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

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

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

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

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

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

En el sistema propuesto solo se utiliza el canal L* correspondiente al canal de luminosidad, permitiendo así reducir el número de canales a procesar. La imagen representada en L* proporciona una mejor variación entre los píxeles claros y oscuros. La imagen obtenida en este paso se denota como como L.

2) Clusterización por K-means

Para la extracción del texto contenido en la imagen L se utilizó el método de clusterización de k means definido por la ecuación 2 [26]:

$$argmin(SSE)$$

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

$$w^{(i,j)} = \begin{cases} 1 & \text{si } x^{(i)} \in \text{cluster } j \\ 0 & \text{en caso contrario} \end{cases}$$

En la ecuación 2, $\mu^{(j)}$ es el centroide del cluster j, SSE es la suma de la estimación cuadrática del error y $w^{(i,j)}$ es una variable indicadora que determina la asignación de $x^{(i)}$ al cluster j. Este método permite agrupar la información de entrada en grupos y encontrar el centroide de los mismos. El número de clusters define cuantos grupos diferentes se van a generar dependiendo del valor de los píxeles. De manera experimental 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 píxeles 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, provocando que segmentación del texto contenga píxeles que no corresponden al texto de la etiqueta; esto afecta directamente la posibilidad del sistema de detectar falsos negativos. 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 con 3 clusters; el cluster extra permite asociar a este grupo los píxeles con valor intermedio entre el cable y el texto. Por lo tanto, en este trabajo se usan 4 clusters para disminuir la posibilidad de que el sistema genere falsos negativos.

A los cada uno de los clusters se les asigno el valor del centroide de los píxeles contenidos en los mismos, permitiendo reducir así el número de valores de intensidad. La imagen obtenida en este paso se denotó como IMG_{kmn} y los valores de los centroides se almacenaron en la lista $Values = [C_1, C_2, C_3, C_4]$, para su uso posterior; en Values los valores se encuentran ordenados por intensidad, de menor a mayor.

Paradeterminar el valor correspondiente al texto en la imagen se propuso la ecuación 3, la cual considera dos casos; cuando el fondo de la imagen es claro, se tiene texto oscuro y en el caso contrario, al tener un fondo de imagen oscuro, se tiene un texto claro.

$$text_{val} = \begin{cases} Values [0] & \text{cuando } \frac{\sum_{i=0}^{10} \sum_{j=0}^{10} L_{chan[i,j]}}{10} \geq 200 \\ Values [3] & \text{en caso contrario} \end{cases} \quad (3)$$

Con el valor correspondiente al texto se realizó una umbralización utilizando el valor de $text_{val}$ definida por la ecuación 4:

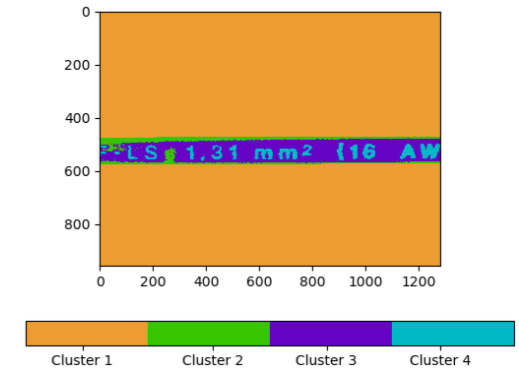


Fig. 2 Resultados del clusterizado, cada grupo está representado con un color distinto

$$mask_{kmn}(x,y) = \begin{cases} 0 & \text{cuando } IMG_{kmn}(x,y) \neq text_{val} \\ 255 & \text{en caso contrario} \end{cases} \quad (4)$$

El segmento que contiene al texto en la imagen original IMG es extraído utilizando la máscara $mask_{kmn}$ a partir de la ecuación 5:

$$IMG_{text}(x,y) = \begin{cases} IMG(x,y) & \text{si } mask_{kmn}(x,y) = 255 \\ 0 & \text{en caso contrario} \end{cases} \quad (5)$$

En la Figura 2 se presenta una representación visual de los distintos grupos obtenidos al utilizar la clusterización por K-means. En la parte inferior se presenta la barra de color que define el cluster, cada uno marcado con un color distinto.

3) Suavizado de la imagen utilizando un filtro de mediana.

El suavizado de la imagen IMG_{text} se realizó utilizando un filtro de mediana [27], permitiendo reducir pequeños puntos de ruido generados por el paso anterior. Este ruido corresponde a los bordes del texto, en donde dependiendo de la calidad de la impresión pueden presentarse bordes borrosos o difuminados y segmentos internos del texto difuminados; esto genera un borde irregular con recorte en el texto. Una de las recomendaciones para el adecuado funcionamiento del motor OCR Tesseract es suavizar la imagen de entrada al OCR [28]. El valor de IMG_{text} se actualiza con los datos de la imagen suavizada.

La Figura 3 presenta un ejemplo del resultado obtenido de la etapa de pre procesamiento utilizando el K-means

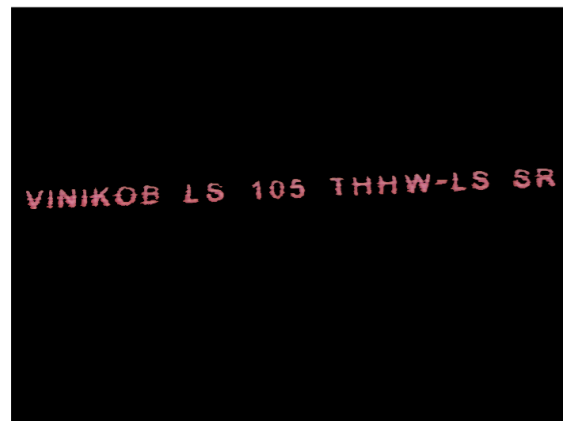


Fig. 3 Resultado de la etapa de pre procesamiento utilizando K-means.

En la Figura 3, es posible observar el resultado de la etapa de pre procesado, la cual permite segmentar el texto correspondiente a la impresión en el cable, removiendo el fondo y el cable de la imagen.

B. Etapa 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 es el motor Tesseract OCR.

La imagen IMG_{text} antes de ser ingresada al motor OCR es umbralizada de manera automática utilizando el método de Otsu [29], debido a que al ser un método de umbralizado automático permite separar la información de IMG_{text} en dos clases. Estas clases se definen en la ecuación 6:

$$IMG_{bin}(x,y) = \begin{cases} 255 & \text{cuando } IMG_{text}(x,y) > thr \\ 0 & \text{en caso contrario} \end{cases} \quad (6)$$

Donde x y y corresponden a las coordenadas de la imagen, thr corresponde al valor límite obtenido del método de Otsu. Por lo tanto, IMG_{bin} corresponde a una imagen binaria en donde los valores mayores a thr se convierten a blanco (255) y los valores menores se convierten a negro (0).

Para mejorar el resultado a obtener del motor OCR se configuraron algunos parámetros del mismo. Existen diferentes formas de analizar la imagen de entrada. Tesseract permite utilizar distintos modos de segmentación de páginas: en una región pequeña, texto contenido en un círculo, texto escaso; dentro de estas opciones se eligió como método de segmentación el

método de segmentación automático, sin OSD (Optical Script Detection, en español detección óptica de escritura/fuente).

A partir de la versión 4.00 de Tesseract se incluye un motor de reconocimiento basado en redes neuronales utilizando LSTM, que ofrece una precisión significativamente mayor en las imágenes con texto, por lo tanto, este modo de trabajo del motor OCR fue seleccionado, utilizando a la par los modelos de lenguaje español e inglés.

Para mejorar el reconocimiento del texto contenido en las imágenes, se deshabilitó el diccionario de Tesseract y se creó una lista blanca con los caracteres a detectar: "0123456789abcdefghijklmnopqrstuvwxyzABCDEF GHIJKLMNOPQRSTUVWXYZ()-. ,[]".

Teniendo los parámetros optimizados para este análisis, la imagen IMG_{bin} se procesa en el motor OCR, obteniendo de él los datos presentados en la Tabla II.

TABLA II. DATOS OBTENIDOS DEL MOTOR OCR

Dato	Descripción
Level	Nivel jerárquico de la palabra en la imagen del 1 al 5, (1 – página, 2 – bloque, 3 – párrafo, 4 – línea, 5 – palabra).
Page_num	El número de página cuando se ingresa más de una imagen.
Block_num	El número de bloque dentro de la página.
Par_num	El número de párrafo dentro del bloque.
Line_num	Número de línea en el párrafo.
Word_num	El número de palabra dentro de la línea.
Left	Coordenada X en pixeles de la esquina superior izquierda del rectángulo delimitador de texto, comenzando desde la izquierda de la imagen.
Top	Coordenada Y en pixeles de la esquina superior izquierda del rectángulo delimitador de texto.
Width	Ancho del rectángulo delimitador del texto en pixeles.
Height	Altura del rectángulo delimitador en pixeles.
Conf	Valor de confianza de 0 a 100 (menor a mayor confianza), -1 para todos los niveles excepto el nivel 5 correspondiente a la palabra.
Text	Texto detectado

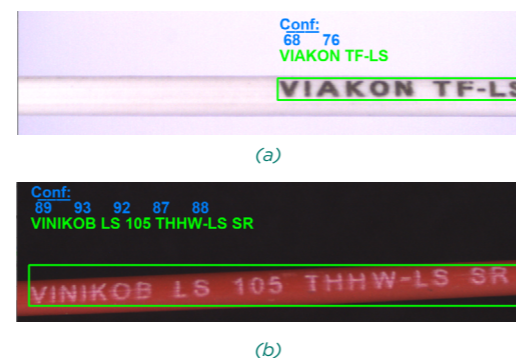


Fig. 4 Resultado del motor OCR. (a) Imagen con fondo claro y texto oscuro; (b) imagen con fondo oscuro y texto claro.

De los datos presentados en la Tabla II, obtenidos como resultado del OCR, se utilizaron: el número de palabras y las coordenadas de los vértices del rectángulo delimitador alrededor de la palabra, calculadas a partir de left, top, width y height. Con estas coordenadas se realiza el cálculo para obtener un rectángulo dentro del cual se encuentran todas las palabras reconocidas; la confianza de la predicción y la predicción del texto de la imagen, la cual está representada como una cadena de caracteres.

La Figura 4 presenta ejemplos visuales de resultados obtenido por el sistema. Se presenta la información resultante del motor OCR, en orden de aparición superior a inferior representan: la certeza de cada una de las palabras; las palabras obtenidas y agrupadas en una sola cadena de texto; y el rectángulo que contiene a todas las palabras.

C. Etapa de validación

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 es el motor Tesseract OCR.

Para validar el texto obtenido por el OCR se desarrolló un algoritmo 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.

Bajo este enfoque el algoritmo se desarrolla alrededor de un vector de ceros, representado en la Ecuación 7:

$$A = [0_0 \ 0_i \ \dots \ 0_{m-1}]_m \quad (7)$$

En donde m corresponde a la cantidad de caracteres contenidos en el texto real de la etiqueta. Cada producto contiene una longitud distinta. Se considera un carácter válido solamente si se encuentra en la posición correcta y el carácter es el mismo que el analizado, como se representa en la Ecuación 8:

$$\text{Si } C = A_i \wedge i_c = i_a \text{ entonces } A_i = 1 \quad (8)$$

En donde C corresponde al carácter a revisar; A_i corresponde al carácter del vector en la posición i , i_c corresponde a la posición del carácter c e i_a corresponde a la posición del carácter en el vector A .

Al finalizar una secuencia de imágenes, los caracteres que no se validaron, se mantienen en cero, representando un error en el etiquetado y permitiendo de esta manera encontrar problemas o defectos en el etiquetado y especificar los caracteres en los cuales se están teniendo problemas.

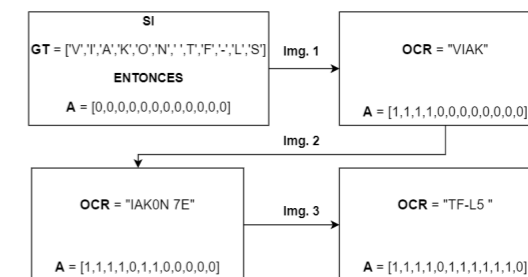


Fig. 5 Ejemplo de la etapa de validación.

En la Figura 5 se presenta un ejemplo de la etapa de validación y la forma en que se va rellenando el vector A. Se presenta un diagrama de bloques en el cual se realiza una pequeña simulación del funcionamiento de la etapa de validación, como entrada se tiene que el ground truth de la etiqueta es "VIKON TF-LS" por lo tanto el vector A forma un vector de ceros de tamaño 12, que es el número de caracteres contenidos en la etiqueta, considerando el espacio. Al procesar una primera imagen se obtiene un resultado del motor OCR, este resultado corresponde al texto en la imagen; en este primer resultado se obtuvo el texto "VIK" que corresponde a los primeros cuatro caracteres de la etiqueta, por tanto, el vector A cambia su valor a 1 en las primeras cuatro posiciones. En la siguiente imagen que es procesada, el resultado obtenido por el OCR contiene tres caracteres erróneos, por lo tanto, a pesar de coincidir con la posición correcta, al no ser el mismo carácter que el de la etiqueta, el vector A mantiene el valor de 0 en esa posición. Finalmente, al procesar una tercera imagen el motor de OCR reconoce bien los caracteres detectados anteriormente de manera errónea, por lo tanto, estos caracteres se pueden validar y el vector cambia a 1 en la posición correspondiente a estos caracteres. Para reducir la posibilidad de generar falsos positivos, se decidió utilizar un número impar de secuencias, combinando el resultado obtenido de una secuencia impar con el resultado de la secuencia impar anterior.

IV. RESULTADOS

A. Base de datos de imágenes

Para realizar la experimentación del sistema prototipo se creó una base de datos formada por diferentes cables. La Tabla III 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.

TABLA III. BASE DE DATOS PARA DESARROLLO DEL ANÁLISIS EXPERIMENTAL

Fabricante	Color del cable	Color del texto	Calibre	Núm. de imágenes	Imágenes con texto	Alias
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
Imágenes totales				909	707	

La base de datos está compuesta por cables que pueden ser similares en cuestión de color, el color del texto, el tipo de cable y el calibre; estas características son similares a las que se usan con mayor frecuencia en Planta. 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, 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. El texto impreso sobre el conductor está formado en su totalidad por diferentes tomas continuas, por lo que para la validación del texto se utiliza una secuencia de imágenes.

B. Etiquetado de imágenes con la ground truth

Para cada una de las imágenes de la base de datos se generó un archivo de ground truth, el cual representa el texto real contenido en la imagen. Esta 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. Por lo que, realizar el análisis o la predicción del OCR sobre texto con segmentos incompletos en algunas letras, se considera una situación fuera de lo esperado para el motor OCR, afectando directamente a la precisión de la detección.

C. Evaluación del desempeño del método

Para evaluar el desempeño del sistema se obtuvieron los aciertos por colección y método. 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 IV presenta los resultados de dicho análisis y en la Figura 6 se presentan gráficamente estos resultados.

TABLA IV. PORCENTAJE DE ERROR EN LAS DISTINTAS COLECCIONES POR SECUENCIA

Conjunt o	Método K-means % error por secuencia ^a							No. de caractere s en etiqueta
	Sec. 1	Sec. 2	Sec. 3	Sec. 4	Sec. 5	Sec. 6	Sec. 7	
Indiana rojo	8.93	3.30	2.56	1.98	1.37	0.73	0.00	91
Viakon blanco	8.94	1.00	0.00	0.00	0.00	0.00	0.00	50
Kobrex azul	51.81	36.75	29.26	24.30	20.24	16.87	13.25	83
Kobrex rojo	8.81	4.73	3.35	2.76	2.44	2.44	2.44	41
Kobrex Vinikob	32.53	14.22	6.16	2.26	0.69	0.40	0.00	83
Viakon negro	12.73	6.19	3.74	2.72	1.95	1.66	1.44	87
Viakon UL	20.22	8.29	6.47	5.39	4.41	3.36	2.70	68
Viakon verde	5.12	1.31	0.80	0.73	0.64	0.54	0.43	87
Prom.	18.64	9.47	6.54	5.02	3.97	3.25	2.53	

^a Una secuencia contiene la totalidad de imágenes con fragmentos de texto, con los cuales se forma la etiqueta del producto.

En la Tabla IV y en la Figura 6 es posible observar que al utilizar un mayor número de secuencias la tasa del error disminuye. El 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.

V. DISCUSIÓN DE LOS RESULTADOS OBTENIDOS

En la experimentación realizada se observó que el sistema inteligente propuesto 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, el sistema 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.

Debido a que la etiqueta que es colocada en el conductor eléctrico está segmentada en un grupo de imágenes, se compararon resultados que utilizan una sola secuencia de imágenes contra un número mayor de secuencias. En esta comparación fue posible observar una disminución en el error; sin embargo, se tiene un incremento en el número de imágenes necesarias para realizar la validación del texto. De igual manera, cuando el texto validado contiene problemas de etiquetado, fue posible observar que el error deja de disminuir a lo largo del incremento de las secuencias.

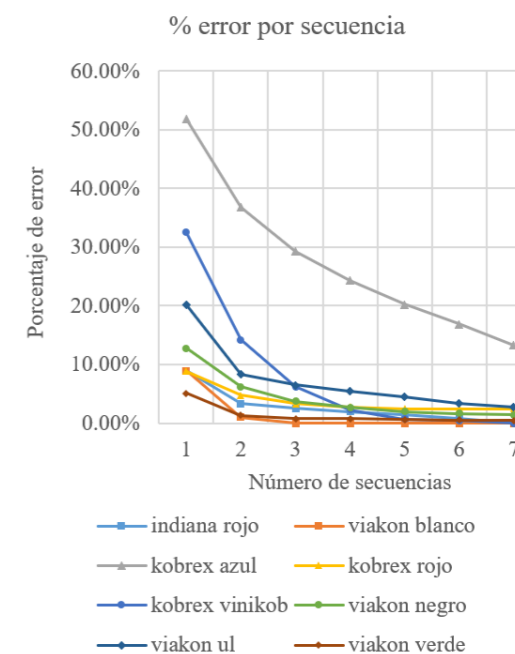


Fig. 6 Porcentaje de error por secuencia en las distintas colecciones.

VI. CONCLUSIONES

En el procesamiento digital de imágenes para la creación del prototipo de sistema inteligente propuesto se empleó el método K-means para agrupar el texto, el cable y el fondo en tres distintos grupos, pero, 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 sistema propuesto 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, presenta desventaja cuando el texto es moderadamente borroso o se presentan tallones en el cable. La etapa de pre procesamiento permitió 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; sin embargo, 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.

En el caso de estudio presentado, la etiqueta que es colocada en el conductor eléctrico está segmentada en un grupo de imágenes; el algoritmo desarrollado permite conocer cuáles caracteres son los que no se pudieron validar a lo largo de la secuencia de imágenes y evitar falsos positivos, ya que los requisitos para que el texto sea validado contra el del ground truth son dos: la posición y la igualdad del carácter a validar. La validación del texto utilizando el sistema propuesto permitió obtener una tasa de error promedio del: 6.54%, 3.97% y 2.53% al utilizar tres, cinco y siete secuencias, respectivamente. Esta validación corresponde al número de etiquetas que el sistema tiene que verificar para determinar si hay un error en la impresión o no, en el proceso de etiquetado.

Como trabajo futuro se sugiere la implementación del algoritmo y realizar las optimizaciones necesarias para poderlo utilizar en un SBC (en inglés, single board computer), buscando implementar el prototipo en planta, sin la necesidad de requerir un equipo de cómputo de escritorio. La paralelización de las etapas permitiría utilizar el sistema de detección y reconocimiento de texto impreso en líneas de alta velocidad. Otra recomendación para realizar trabajo futuro es diseñar una interfaz gráfica que permita mostrar información detallada sobre el reconocimiento y la detección del texto, presentando la información en un reporte

digital y un explorador o visualizador de las imágenes con problemas. También se sugiere como trabajo futuro diseñar un algoritmo que integre la etapa de pre procesamiento y de OCR utilizando otras técnicas de visión, por ejemplo: operaciones morfológicas, umbralización adaptativa; técnicas de machine learning; entrenamiento de clasificadores en cascada, clasificadores SVM, arboles de decisiones; sin dejar de lado también la posibilidad de utilizar técnicas de aprendizaje profundo o transferencia de aprendizaje, aplicado para el caso en específico de la detección de texto en conductores eléctricos.

VII. REFERENCIAS

- [1] SAMPSTEMI, "Cable Extrusion," [Online]. Available: <https://www.sampsistemi.com/cable-extrusion/> [accessed Oct. 20, 2020].
- [2] "Wire and cable marking machines." [Online]. Available: <https://www.schleuniger.com/en-us/products/peripherals/marking-/-printing-/-labeling/> [accessed Oct. 20, 2020].
- [3] "Coding & Marking Printers | Gem Gravure." [Online]. Available: <https://www.gemgravure.com/coding-marking-printers/> [accessed Oct. 20, 2020].
- [4] "Impresoras para tubos y cables | Videojet México." [Online]. Available: <https://www.videojet.mx/mx/homepage/industry-solutions/wire-cablepipe.html> [accessed Oct. 20, 2020].
- [5] A. K. Bhunia, G. Kumar, P. P. Roy, R. Balasubramanian, and U. Pal, "Text recognition in scene image and video frame using Color Channel selection," *Multimed. Tools Appl.*, vol. 77, no. 7, pp. 8551-8578, Apr. 2018, doi: 10.1007/s11042-017-4750-6.
- [6] S. M. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong, and R. Young, "ICDAR 2003 robust reading competitions," in *Seventh International Conference on Document Analysis and Recognition*, 2003. Proceedings., Aug. 2003, pp. 682-687, doi: 10.1109/ICDAR.2003.1227749.
- [7] N. V. Rao et al., "Optical character recognition technique algorithms," vol. 16, no. 2, pp. 275-282, 2016.
- [8] L. von Ahn, M. Blum, N. J. Hopper, and J. Langford, "CAPTCHA: Using Hard AI Problems for Security," in *Advances in Cryptology - EUROCRYPT*, pp. 294-311, 2003.
- [9] J. B. Pedersen, K. Nasrollahi, and T. B. Moeslund, "Quality inspection of printed texts," in *2016 International Conference on Systems, Signals and Image Processing (IWSSIP)*, Bratislava, Slovakia, May 2016, pp. 1-4, doi: 10.1109/IWSSIP.2016.7502718.
- [10] F. De Sousa Ribeiro et al., "An End-to-End Deep Neural Architecture for Optical Character Verification and Recognition in Retail Food Packaging," in *2018 25th IEEE International Conference on Image Processing (ICIP)*, Athens, Oct. 2018, pp. 2376-2380, doi: 10.1109/ICIP.2018.8451555.
- [11] E. Shelhamer, J. Long, and T. Darrell, "Fully Convolutional Networks for Semantic Segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640-651, Apr. 2017, doi: 10.1109/TPAMI.2016.2572683.
- [12] D. Chen, J.-M. Odobez, and H. Bourlard, "Text detection and recognition in images and video frames," *Pattern Recognit.*, vol. 37, no. 3, pp. 595-608, Mar. 2004, doi: 10.1016/j.patcog.2003.06.001.
- [13] K. Messer, J. Kittler, and W. Christmas, "Automatic Sports Classification," in *Pattern Recognition, International Conference on, Los Alamitos, CA, USA*, Aug. 2002, vol. 2, pp.1005-1008, doi: 10.1109/ICPR.2002.1048475.
- [14] J. C. Rodríguez-Rodríguez, A. Quesada-Arencibia, R. Moreno-Díaz, and C. R. García, "A Character Segmentation Proposal for High-Speed Visual Monitoring of Expiration Codes on Beverage Cans," *Sensors*, vol. 16, no. 4, Apr. 2016, doi: 10.3390/s16040527.
- [15] W. Q. Khan and R. Q. Khan, "Urdu optical character recognition technique using point feature matching; a generic approach," in *2015 International Conference on Information and Communication Technologies (ICICT)*, Dec. 2015, pp. 1-7, doi: 10.1109/ICICT.2015.7469576.
- [16] M. A. H. Monil, M. S. Q. Z. Nine, B. Poon, M. A. Amini, and H. Yan, "Bangla text processing and recognition based on Fuzzy unsupervised Feature Extraction and SVM," in *2013 International Conference on Machine Learning and Cybernetics*, Jul. 2013, vol. 03, pp. 1272-1278, doi: 10.1109/ICMLC.2013.6890784.
- [17] T. Hassan and H. A. Khan, "Handwritten Bangla numeral recognition using Local Binary Pattern," in *2015 International Conference on Electrical Engineering and Information Communication Technology (ICEEICT)*, May 2015, pp. 1-4, doi: 10.1109/ICEEICT.2015.7307371.
- [18] N. Das, R. Sarkar, S. Basu, M. Kundu, M. Nasipuri, and D. K. Basu, "A Genetic Algorithm Based Region Sampling for Selection of Local Features in Handwritten Digit Recognition Application," *Appl Soft Comput*, vol. 12, no. 5, pp. 1592-1606, May 2012, doi: 10.1016/j.asoc.2011.11.030.
- [19] S. Pasha and M. C. Padma, "Handwritten Kannada character recognition using wavelet transform and structural features," in *2015 International Conference on Emerging Research in Electronics, Computer Science and Technology (ICERECT)*, Dec. 2015, pp. 346-351, doi: 10.1109/ERECT.2015.7499039.
- [20] C. Liyanage, T. Nadungodage, and R. Weerasinghe, "Developing a commercial grade Tamil OCR for recognizing font and size independent text," in *2015 Fifteenth International Conference on Advances in ICT for Emerging Regions (ICTer)*, Aug. 2015, pp. 130-134, doi: 10.1109/ICTER.2015.7377678.
- [21] A. Singh and S. Desai, "Optical character recognition using template matching and back propagation algorithm," in *2016 International Conference on Inventive Computation Technologies (ICICT)*, Aug. 2016, vol. 3, pp. 1-6, doi: 10.1109/INVENTIVE.2016.7830161.
- [22] GitHub, "tesseract-ocr/tesseract," [Online]. Available: <https://github.com/tesseract-ocr/tesseract> [accessed Oct. 20, 2020].
- [23] G. Van Rossum and F. L. Drake, *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace, 2009.
- [24] G. Bradski, "The OpenCV Library," Dr Dobbs J. Softw. Tools, 2000.
- [25] "OpenCV: Color conversions," [Online]. Available: https://docs.opencv.org/3.4/de/d25/imgproc_color_conversions.html#color_convert_rgb_lab [accessed Oct. 20, 2020].
- [26] S. Raschka and 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.
- [27] A. Kaehler and G. R. Bradski, *Learning OpenCV 3: computer vision in C++ with the OpenCV library*, First edition, Second release. Sebastopol, CA: O'Reilly Media, 2017.
- [28] "Improving the quality of the output," tessdoc. <https://tesseract-ocr.github.io/tessdoc/ImproveQuality.html> (accessed Oct. 20, 2020).
- [29] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," *IEEE Trans. Syst. Man Cybern.*, vol. 9, no. 1, pp. 62-66, Jan. 1979, doi: 10.1109/TSMC.1979.4310076.

AUTHORS



Juan J.
Navarro

Es ingeniero de proyectos de automatización en Continuous Improvement Managment Associates S. C. Obtuvo su título de Ingeniero en Mecatrónica en 2012 en la Universidad Autónoma de San Luis Potosí (UASLP), San Luis Potosí, México. Actualmente se encuentra estudiando la Maestría en Manufactura Avanzada en CIATEQ A. C. en el campus San Luis Potosí, México. Sus áreas de interés personales son la visión por computadora y las redes neuronales enfocadas a distintas aplicaciones industriales.

Carolina
Reta



Es catedrática CONACYT en CIATEQ A. C. Hidalgo, México, en el departamento de TI, Electrónica y Control, y es miembro del Sistema Nacional de Investigadores Nivel 1 en México. Obtuvo la maestría y el doctorado en Ciencias Computacionales en 2009 y 2014, respectivamente, por el Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE), Puebla, México. Sus áreas de investigación son visión por computadora, realidad aumentada, minería de datos e inteligencia artificial. Su investigación se centra en el desarrollo de métodos computacionales innovadores para los campos de la medicina y la ingeniería de manufactura.