

Desarrollo de un algoritmo de sistema de visión artificial para identificar defectos físicos en latas de atún redondas

Development of a machine vision system algorithm to identify physical defects in round tuna cans

Desenvolvimento de um algoritmo de sistema de visão mecânica para identificar defeitos físicos em latas redondas de atum

Zevallos Tapia, Carlos Manuel¹

<https://orcid.org/0000-0002-0439-1155>

Montes Chang, Rosa Mariel²

<https://orcid.org/0000-0002-9251-0328>

Montes Portella, Carmen Milagros³

<https://orcid.org/0000-0002-0061-3056>

Nolasco Blas, Pamela Mairely⁴

<https://orcid.org/0000-0003-2857-1641>

Bolaños Narciso, Anabel Marilu⁵

<https://orcid.org/0009-0005-8950-3775>

Recibido: 28.01.2023

Aceptado: 28.05.2023

RESUMEN

Se desarrolló un algoritmo destinado a la identificación de los defectos físicos en las latas de atún redondas aplicando técnicas de visión artificial y procesamiento de imágenes. El algoritmo centra su funcionamiento en un ordenador de 12 Gb de RAM y Windows 10 de 64 bits, que conectado a una cámara de 108 MP, mediante el uso de la aplicación DroidCam, puede adquirir imágenes de la parte lateral de las latas de atún redondas. Estas imágenes adquiridas son procesadas dentro del ordenador. Se usó la técnica de visión artificial y métodos estadísticos para identificar los defectos físicos en la parte lateral de las latas de atún redondas, tales como, golpes pequeños, grandes y más de un solo golpe. Tras llevar a cabo pruebas que se realizaron con el algoritmo, se obtuvo como resultado una precisión del sistema del 96.08%. Se concluyó que, la implementación de este algoritmo asegura la calidad de las latas de atún redondas, así como facilita un eficaz control en los procesos de la industria al detectar aquellas latas que presentan defectos físicos.

Palabras clave: Defectos físicos, Procesamiento de imágenes, Latas de atún redondas.

ABSTRACT

An algorithm was developed for the identification of physical defects in round tuna cans by applying computer vision and image processing techniques. The algorithm focuses its operation on a computer with 12 Gb of RAM and Windows 10 64 bits, which connected to a 108 MP camera, by using the DroidCam application, can acquire images of the side of the round tuna cans. These acquired images are processed inside the computer. Machine vision technique and statistical methods were used to identify physical defects on the side of round tuna cans, such as, small, large and more than one hit. After carrying out tests that were performed with the algorithm, a system accuracy of 96.08% was obtained as a result. It was concluded that the implementation of this algorithm ensures the quality of round tuna cans, as well as facilitates effective control in industry processes by detecting those cans that present physical defects.

Keywords: Physical defects, Image processing, Round tuna cans.

¹ Real Mix S.A.C. Lima-Surquillo. Perú. Bachiller. carloszevallostapia2020@gmail.com

² Multiservicio y Corporación M&C E.I.R.L Lima-Ate. Perú. Bachiller. monteschangr@gmail.com

³ Universidad Cesar Vallejo. Chimbote. Perú. Bachiller. carmenmontes416@gmail.com

⁴ Universidad Cesar Vallejo. Chimbote. Perú. Bachiller. pam28082002@gmail.com

⁵ Servicio Nacional de Sanidad Agraria. Lima. Perú. Magister. anabelbolanosn@gmail.com

RESUMO

Foi desenvolvido um algoritmo para a identificação de defeitos físicos em latas redondas de atum por meio da aplicação de técnicas de visão computacional e processamento de imagens. O algoritmo é baseado em um computador com 12 Gb de RAM e Windows 10 de 64 bits, que conectado a uma câmera de 108 MP, usando o aplicativo DroidCam, pode adquirir imagens da lateral de latas de atum redondas. Essas imagens adquiridas são processadas dentro do computador. A técnica de visão mecânica e os métodos estatísticos foram usados para identificar defeitos físicos na lateral das latas redondas de atum, como pequenos, grandes e mais de uma batida. Após a realização de testes com o algoritmo, foi obtida uma precisão do sistema de 96,08%. Concluiu-se que a implementação deste algoritmo garante a qualidade das latas redondas de atum, bem como facilita o controle eficaz nos processos da indústria ao detectar as latas que apresentam defeitos físicos.

Palavras-chave: defeitos físicos, processamento de imagens, latas redondas de atum.

Introducción

En la actualidad, el Perú tiene una alta disponibilidad del recurso pesquero y las grandes industrias procesan, en latas, grandes cantidades de especies marinas como, por ejemplo, el atún. Sin embargo, un problema persistente ha afectado a estas empresas industriales durante años: los defectos físicos presentes en las latas. Dentro de los procesos suelen suceder diversos tipos de sucesos que conllevan a defectos físicos como golpes o imperfecciones que dañan el contenido, ocasionando enfermedades para aquellos que lo ingieran en esas condiciones. En este sentido, es sustancial que podamos definir algunos términos que se tratarán en el presente estudio. La visión artificial, es una materia de la inteligencia artificial que busca dotar a las máquinas de la capacidad de comprender, analizar y procesar información visual con una gran precisión (i García, 2012). Para poder realizar este sistema de visión artificial utilizaremos el lenguaje de programación Python que es un lenguaje de programación de alto nivel muy utilizado en el campo artificial para desarrollar diversas tareas Severance et al. (2009). De este modo, estas dos herramientas van ayudar a cumplir con nuestro propósito, de esta manera los defectos físicos como abolladuras o golpes en las latas son perjudiciales para el contenido interior y posteriormente para la salud de los consumidores (Munoz, 2023). Por eso, es importante hacer hincapié en este problema, porque en el Perú el enlatado de atún es una actividad grande, por lo que, la producción y exportación de latas de atún es una labor económica significativa en el Perú. Si estas latas presentan imperfecciones, las empresas disminuyen sus ventas, pierden credibilidad y tendrían sanciones legales, en otras palabras, pierde mucho dinero. De la misma manera, si no se tiene un buen control de calidad de envase, la empresa no va a poder comercializar un buen producto ni mucho menos exportarlos (Otarola, 2014). Teniendo en cuenta los problemas sucedidos en el país, problemas de este tipo les ha sucedido a empresas reconocidas como Plaza Vea y Tottus, quienes, entre otros productos, comercializan latas de atún. INDECOPI (El Instituto Nacional de Defensa de la Competencia y de la Protección de la Propiedad Intelectual) sancionó a estas empresas con una multa total de 15 UIT debido a la venta de latas de atún con abolladuras y golpes (Sanchez, 2023). De igual manera, en Chimbote se retiraron 40 cajas de latas de atún que tuvieron imperfecciones y de dudosa procedencia en el mercado la Perla (Diario de Chimbote, 2020). En este sentido, distintos autores investigaron y aplicaron la visión artificial para solucionar un problema parecido al nuestro. En primer lugar, un estudio desarrolló un sistema de visión artificial usando redes neuronales con el objetivo de identificar defectos o fallas en el codificado de las latas de atún, donde su resultado fue que la eficiencia de su modelo no llega al 85% de efectividad (Palma, 2020). De manera similar, en otra investigación, se realizó un proyecto de detección de imperfecciones de latas de refrescos mediante visión artificial y las redes neuronales, en donde tuvieron resultados muy satisfactorios (98.17%) (Poblet, 2022). En otra investigación, se propuso una mejora para detectar golpes que presenten las latas de atún mediante el uso de visión artificial, logrando que el prototipo funcione correctamente detectando abolladuras y golpes en las latas (Alvarado, 2021). En base a las investigaciones mencionadas nos podemos dar cuenta que el proyecto es posible de realizar e implementar, ya que la visión artificial va evolucionando según la tecnología avance, lo cual se recomienda a las empresas tener alta adaptabilidad para no quedarse atrás en la utilización de estas herramientas tecnológicas (Dominguez, 2021). La adaptabilidad tiene que ser una característica innata dentro de las empresas que no quieren irse a la quiebra (Caravedo, 2016). Un término que es parecido a la visión artificial es la inteligencia artificial, que ha comenzado a surgir en este último año, pues, la inteligencia artificial se relaciona a la capacidad de máquinas para hacer tareas (Soria et al., 2022). Ahora, las empresas necesitan de estas herramientas innovadoras, las nuevas tecnologías ofrecen a las empresas sumar eficiencia y reducir costos (Mendoza, 2023). Por lo tanto, este

tipo de investigaciones hace que las empresas puedan darse cuenta que pueden mejorar sus métodos, a través de investigaciones innovadoras pueden conducir al descubrimiento de nuevas tecnologías que las favorecen (Romero, 2022).

El proyecto presentado, surge como una respuesta crucial a los desafíos que enfrenta la industria del enlatado de atún en el Perú, abordando no solo una problemática crítica y urgente, sino que también se presenta como una posibilidad para mejorar la seguridad de los consumidores, fortalecer la industria del enlatado de atún y promover la adopción de tecnologías avanzadas en el entorno empresarial peruano. Este proyecto contribuye a la seguridad alimentaria, a través de la detección temprana de defectos físicos de las latas de atún redondas a través de la visión artificial. El objetivo general del proyecto es desarrollar un algoritmo de visión artificial que permita identificar los defectos físicos de la lata de atún redonda. Como objetivos específicos tenemos, implementar un ambiente controlado para que el sistema de visión artificial trabaje correctamente, aplicar las diferentes técnicas de visión artificial en el programa Python para identificar los defectos físicos en las latas de atún redondas.

Material y métodos:

El presente estudio tiene un enfoque de desarrollo modular, donde la aplicación se construye a partir de múltiples componentes o módulos independientes. Cada componente se desarrolla por separado, lo que nos permite abordar el proyecto de manera incremental. Para llevar a cabo esta metodología, utilizamos un entorno controlado y programamos con IDLE (Python 3.12.0). Los implementos utilizados en la elaboración incluyen: Luces LED 9 watts de 1 metro de largo, una estructura cerrada de 20 x 20 x 20 cm., ordenador de 12 GB de RAM, batería DURACELL 9 V, cámara de 108 MP. Las luces LED 9 watts tiene 800 lumens que es la cantidad de luz que genera para que se visualice mejor el objeto, que, en este caso es una lata de atún redonda.

Para capturar las imágenes que son almacenadas en carpetas creadas por el propio programa Python con el propósito de utilizarlas posteriormente, es necesario diseñar una estructura de precisión como se puede observar en la figura 1. Esta estructura facilitará la adquisición de imágenes utilizando el método propuesto en un entorno controlado, donde se regulan el movimiento de la cámara y la iluminación. La fuente de luz debe ubicarse lateralmente, es decir, al lado del objeto que se va a fotografiar, en este caso tenemos un ambiente de una estructura que tiene 4 lados, por lo cual las luces van a ir puestas en cada uno de esos lados para que se asegure de tener la mejor iluminación. La elección de la iluminación lateral y frontal es la adecuada para acentuar los golpes en una dirección específica, ya que realza los detalles superficiales de los objetos, incluso si son muy pequeños, generando sombras muy definidas. El propósito de esta iluminación es permitir un análisis detallado de la estructura externa de la lata de atún. Por otro lado, hemos dispuesto una distancia de 6 cm entre la fuente de luz y el objeto y la distancia entre la cámara y el objeto es de 13 cm. Esto se debe a que esta distancia entre la cámara y el objeto permite que la cámara enfoque adecuadamente las grietas presentes en la lata de atún.

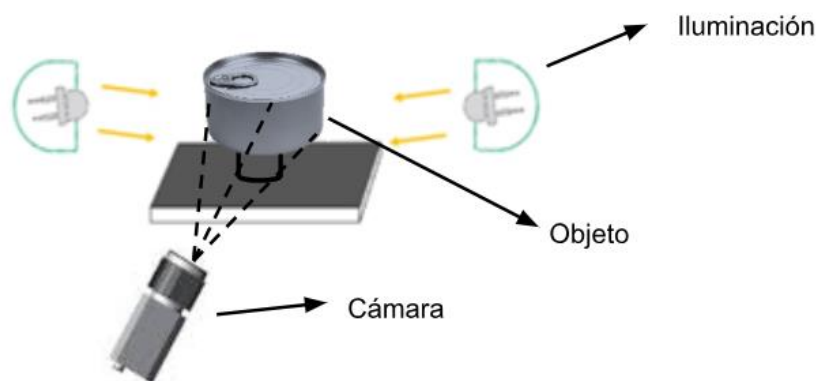


Figura 1. Sistema de iluminación controlado para examinar las muestras de latas de atún redondas

El programa se implementó utilizando como medio de desarrollo el IDLE (Python 3.12.0), que, es un editor que interactúa con Python de forma conveniente, permite escribir y ejecutar scripts, y depurar

código de manera interactiva, proporcionando una ventana de Shell de Python y un editor de texto en un solo entorno (Challenger-Pérez et al., 2014). El utilizado en el presente trabajo fue Python 3.12.0, que permite la implementación de diversas funciones en el lenguaje de la programación que son fáciles de leer y comprender, al ser un programa libre es compatible para todos los sistemas operativos (Beuzen y Timbers, 2022). Las librerías principales que se utilizaron fueron: Open CV versión 4.8.1., que, desarrolla funciones para capturar y manejar imágenes (Rodríguez, 2021). NumPy 1.26.0., que, esta librería permite efectuar operaciones matemáticas sofisticadas, realizar manipulaciones en matrices y arreglos multidimensionales con facilidad (Cardellino, 2021), OS (Operating System) que, es una herramienta que te permite interactuar con el sistema operativo de tu computadora desde tu programa Python (Rico-Schmidt, 2019). Esto significa que puedes realizar tareas como la gestión de archivos. Imutils 0.5.4, que, permite realizar acciones como cambiar el tamaño de imágenes, rotarlas, aplicar recortes, ajustar colores y trabajar con ellas de una manera más amigable y conveniente ((Rosebrock, 2015). Estas funciones que permite Python (3.12.0) nos facilitó la aplicación de las distintas técnicas necesarias para realización del algoritmo.

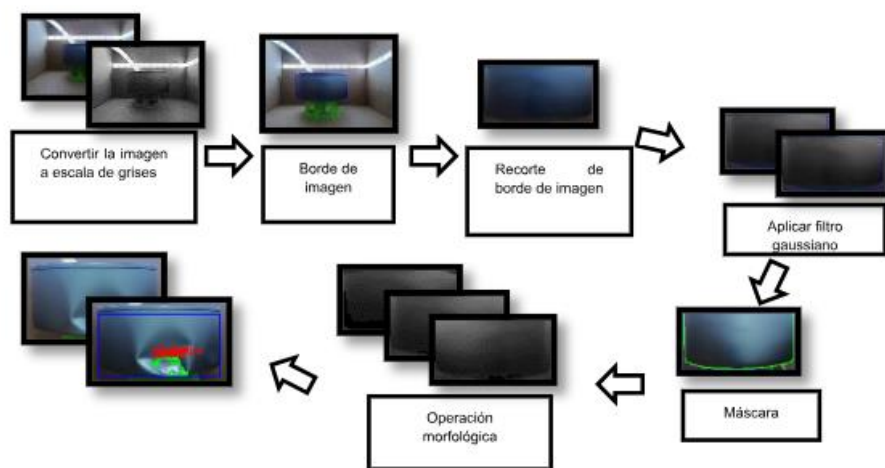


Figura 2. Funcionamiento del algoritmo

En la Figura 2, mostramos cómo el algoritmo procede a aplicar varias técnicas de visión artificial de manera progresiva. Este proceso permite visualizar las acciones realizadas por el algoritmo, y al final, se presenta la imagen de la lata con las imperfecciones visibles. A continuación, detallaremos cada fase del algoritmo. En primer lugar, llevamos a cabo la conversión de la imagen a una escala de grises en donde se asignan valores a cada una de las variables R, G y B como se observa en la ecuación 1. De esta manera, logramos simplificar la imagen, lo que facilita un procesamiento más cómodo y efectivo al disminuir la cantidad de información que necesita ser analizada (Aguilar, 2017).

$$\text{Gray}=0.2989 \times R+0.5870 \times G+0.114 \times B$$

Donde, R, G y B son los distintos valores para los colores rojo, verde y azul de cada pixel, respectivamente. Luego, utilizamos la función detección de objetos mediante marcado con un rectángulo para identificar los bordes de las latas de atún redondas en la imagen. Es una técnica versátil que se utiliza para resaltar, identificar y delimitar áreas de interés en una imagen (Diaz, 2023). Definimos un rectángulo utilizando las coordenadas (x1, y1) y (x2, y2) en el fotograma actual del video. Este rectángulo se dibuja alrededor de una región específica del video. El rectángulo se dibuja utilizando cv2.rectangle() con un color (en este caso, (255,0,0) representa azul) y un grosor de línea de 2 píxeles.

$$\left. \begin{array}{l} x1, y1=157,200 \\ x2, y2=420,342 \end{array} \right\} \quad (2) \text{ Coordenadas}$$

$$\text{cv2.rectangle}(\text{frame}, (x1,y1),(x2,y2),(255,0,0),2)$$

Siguiendo con el procedimiento, ahora los bordes se van a aislar en una región específica del objeto en el fotograma original sin afectar el fotograma en sí y se mostrará en una ventana la lata de atún redonda recortada por los bordes.

$$\text{Objeto} = \text{imAux}[y1:y2,x1:x2]$$

Luego, aplicamos un filtro Gaussiano, el cual, es importante porque se utiliza para suavizar la región de interés (ROI) de la imagen en escala de grises antes de procesarla o realizar análisis adicional (Ortiz et al., 2017), observar en la ecuación 3. Esto se hace con el propósito de reducir el ruido en la imagen y facilitar la detección o el análisis de características específicas.

$$G(X, Y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}} \quad (3)$$

$$\text{blurred} = \text{cv2.GaussianBlur}(\text{gray}, (5, 5), 0)$$

Donde: $G(x, y)$ representa el peso asignado a cada píxel en las coordenadas (x, y) , siendo σ la desviación estándar de la distribución gaussiana y la constante matemática Euler. Para la máscara se utilizamos la detección de bordes Canny, para poder identificar los bordes de la lata de atún redonda en la imagen procesada. Es una técnica ampliamente utilizada en procesamiento de imágenes para identificar los bordes de objetos y características en una imagen (Rebaza, 2017). Para poder ser más específicos en los bordes, hemos colocado los valores de $\text{umbral1} = 18$ y $\text{umbral2} = 100$. Esto es con la finalidad de aislar la lata de atún redonda y no detecte otros objetos que pueda estar en la imagen.

Ahora bien, después de aplicar Canny, se procede a aplicar una operación morfológica considerando `morphologyEx`. Este último, es una técnica utilizada en procesamiento de imágenes para realizar transformaciones morfológicas, como la erosión se puede ver en la ecuación 4 y la dilatación se observa en la ecuación 5, que permiten modificar la forma y la estructura de objetos en una imagen. La dilatación resalta áreas más brillantes y rellenando pequeños espacios (Mehmood, 2022)

Erosión:

$$D(x, y) = \bigcap_{i=-a}^a \bigcap_{j=-b}^b (I(x+i, y+j) \cap K(i, j))$$

Dilatación:

$$D(x, y) = \bigcup_{i=-a}^a \bigcup_{j=-b}^b (I(x+i, y+j) \cup K(i, j))$$

Donde, $I(x, y)$ es el valor del píxel en la imagen original en la posición (x, y) , $K(i, j)$ es el elemento estructurante en la posición (i, j) , $D(x, y)$ es el resultado de la operación morfológica en la posición (x, y) y a y b son los tamaños del elemento estructurante en las direcciones x e y , respectivamente.

En última instancia, identificamos los bordes en la imagen de la lata y los marcamos en la imagen original mediante líneas verdes. Después, agregamos el texto "Golpe" en los bordes marcados. Esto facilita la visualización de los bordes de la lata de atún en la imagen original, lo que ayuda a destacar las áreas anómalas con claridad.

```
if area > 20 and area < 200:
    x, y, w, h = cv2.boundingRect(contour)
    cv2.putText(frame, "Golpe", (x + x1, y + y1 - 10),
                cv2.FONT_HERSHEY_SIMPLEX, 0.8, (0, 0, 255), 2)
```

Luego de dar a conocer el funcionamiento del algoritmo de manera detallada, se dará a conocer la arquitectura del algoritmo (Figura 3), que, da a entender de manera resumida cómo funciona el sistema diseñado.

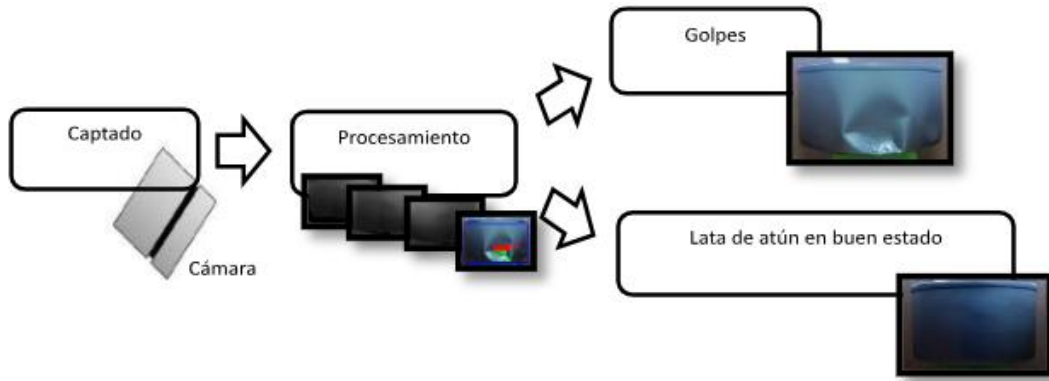


Figura 3. Arquitectura del algoritmo

De acuerdo a la Figura 3, el algoritmo primero capta la imagen de la lata de atún, a través de la cámara, luego comienza hacer todo el procesamiento que se describió en la Figura 2, por último, si es que las latas de atún redondas tienen golpes saldrán los bordes identificados de color verde y un mensaje indicando que hay golpes indicándolos de color rojo, como se puede ver en las Figuras 5-7, por lo que debe descartarse, ya que no está en buenas condiciones. Por lo contrario, si la lata está en buenas condiciones, no saldrá ningún mensaje y se puede comercializar.

Tabla 1.
Función principal

Función	Descripción
<code>x1, y1=157,200</code> <code>x2, y2=420,342</code> <code>cv2.rectangle(frame,</code> <code>(x1,y1),(x2,y2),(255,0,0),2)</code>	Su propósito es realizar un polígono, en este caso un rectángulo, para poder demarcar el área en donde se quiere identificar marcas o daños en la lata de atún redondas. Este rectángulo aparecerá de color azul.

La función mencionada de la Tabla 1 indica que las coordenadas (x1, y1) y (x2, y2) definen la posición del vértice superior izquierdo y del vértice inferior derecho del rectángulo, respectivamente. Creando así, un rectángulo de color azul con un contorno de 2 píxeles de grosor.

Tabla 2.
Función principal

Función	Descripción
<code>umbral1 = 18</code> <code>umbral2 = 100</code> <code>edges = cv2.Canny(morph, umbral1,</code> <code>umbral2)</code> <code>cv2.imshow('canny',edges)</code>	Su propósito es identificar marcas o daños en la lata de atún redondas mediante la configuración de umbrales de 18 (inferior) y 100 (superior), permitiendo la detección de los bordes o golpes en la superficie.

En la Tabla 2, utilizamos la función Canny a una imagen previamente procesada llamada `morph`, estableciendo dos umbrales, `umbral1` y `umbral2`, con valores de 18 y 100 respectivamente, para determinar qué píxeles se considerarán como bordes.

Tabla 3.
Función principal

Función	Descripción
<code>cv2.findContours(edges,</code> <code>cv2.RETR_TREE,</code> <code>cv2.CHAIN_APPROX_SIMPLE)</code>	Esta función se emplea para detectar los golpes en las latas de atún redondas en una imagen binaria. Los parámetros incluyen la imagen binaria, el modo de recuperación de los contornos externos y el método de aproximación de los contornos.

En la Tabla 3, la función sirve para encontrar los contornos en una imagen en blanco y negro, los contornos se estructuran en un árbol jerárquico y simplifica los contornos eliminando puntos redundantes.

Tabla 4.

Función principal

Función	Descripción
<code>cv2.drawContours(frame, [contour], 0, (0, 255, 0), 2)</code>	Esta función se utiliza para dibujar los golpes detectados. Sus argumentos incluyen la imagen donde se desean dibujar los contornos, todos los contornos que se van a dibujar, el color del contorno y el grosor de las líneas utilizadas en el dibujo.

En la Tabla 4, podemos observar que la función representa los contornos a dibujar en una imagen, en donde contiene un parámetro “0” que significa que debe mostrar todos los contornos de la lista. Además, que se representará en color verde con 2 píxeles de grosor.

Tabla 5.

Función principal

Función	Descripción
<code>cv2.FONT_HERSHEY_SIMPLEX, 0.8, (0, 0, 255), 2)</code>	Esta función se utiliza para definir el tipo de fuente que se debe utilizar al dibujar texto en una imagen. Esta constante define un tipo de fuente simple y sans-serif que es fácilmente legible.

En la Tabla 5, observamos que esta función añade un texto a una imagen utilizando una fuente específica, tamaño, color y grosor de línea, lo que nos fue útil para anotar información en la imagen representada.

Resultados

Para dar a conocer la eficiencia de nuestro trabajo se ha utilizado la matriz de confusión. Tuvimos que realizar 100 tomas o fotografías, en donde la cantidad de tomas positivas y tomas negativas a utilizar en la matriz de confusión deben ser iguales. En este caso, trabajaremos con 50 tomas positivas y 50 tomas negativas.

	Positivo	Negativo
Positivo	Verdadero positivo (49)	Falso negativo (1)
Negativo	Falso positivo (2)	Verdadero negativo (48)

Figura 4. Matriz de confusión

Donde:

- Verdadero positivo: Detección correcta de latas en buena condición.
- Falso negativo: Detección incorrecta (no acertada) de latas en buena condición.
- Falso positivo: Detección incorrecta de latas en mala condición.
- Verdadero negativo: Detección incorrecta (no acertada) de latas en mala condición.

Como podemos ver en la Figura 4, los verdaderos positivos de acuerdo a las tomas son cuarenta y nueve y solo se tuvo un falso negativo. Por lo contrario, la cantidad de falso positivos son dos y la cantidad de verdaderos negativos son cuarenta y ocho. Ahora, para poder hallar la eficiencia vamos a utilizar las siguientes ecuaciones

$$Accuracy = \frac{Predicciones\ correctas}{Total\ de\ predicciones} = \frac{97}{100} = 97\%$$

$$Precision = \frac{Verdaderos\ positivos}{Verdaderos\ positivos + Falsos\ positivos} = \frac{49}{49 + 2} = 96.08\%$$

$$\text{Recall} = \frac{\text{Verdaderos positivos}}{\text{Verdaderos positivos} + \text{Falsos negativos}} = \frac{49}{49 + 1} = 98\%$$

$$\text{F1 score} = \frac{2}{\frac{1}{\text{precision}} + \frac{1}{\text{recall}}} = \frac{2}{\frac{1}{0.9608} + \frac{1}{0.98}} = 97.03\%$$

Donde:

- Accuracy: Mide la capacidad general del modelo para predecir correctamente ambas clases.
- Precision: Mide la exactitud de las predicciones positivas del modelo.
- Recall: Mide la capacidad del modelo para identificar todos los casos positivos
- F1 score: Presenta balance entre precisión y Recall.

Por lo tanto, la precisión de nuestro modelo es de 96.08% con un error del 3.92%.

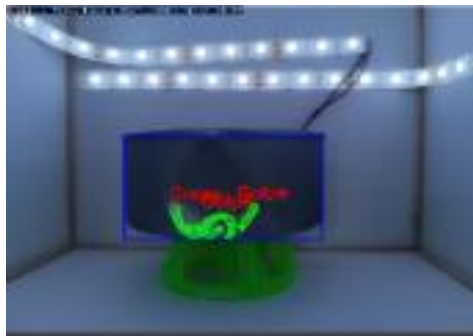


Figura 5. Muestra de lata con un solo golpe grande.

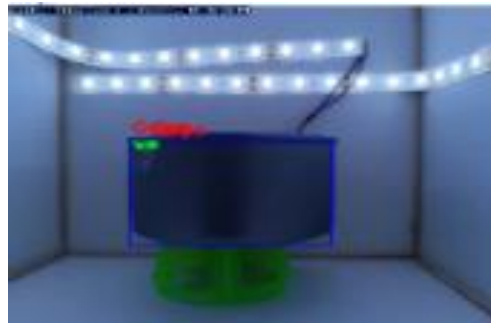


Figura 6. Muestra de lata con un solo golpe pequeño.

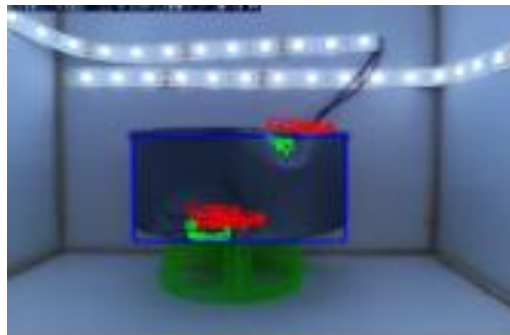


Figura 7. Muestra de lata con dos golpes

Discusión

En la investigación de (Palma, 2020), desarrollo de un sistema de visión artificial usando redes neuronales con el objetivo de identificar defectos o fallas en el codificado de las latas de atún. Tiene como eficiencia un 84% y un margen de error de 16%. En cambio, nuestro proyecto tuvo un 96.08% de precisión como se puede ver en la ecuación 7, esto también debido a que utilizamos un algoritmo de visión artificial aplicando técnicas como se ve en la Figura 2 y no redes neuronales, en esta última se le

exigen mínimo 1000 imágenes. A pesar de que en la investigación entrenaron a la red neuronal con 4000 imágenes tuvieron un porcentaje de precisión menor al 90%, lo cual no es aceptable.

De igual manera, en la investigación de (Poblet, 2022), propone un sistema que constaba de detectar los defectos que hay en los paquetes de latas de refrescos, en este caso, si los refrescos están volteados o con algún defecto configurado debe detectarlo. Su eficiencia es de 98.17% en donde utilizaron 6000 imágenes para entrenar a esta red neuronal. Mientras que, nuestro proyecto tiene un 96.08% de precisión, en este caso, nuestro proyecto tiene un porcentaje menor, pero, los datos que hallaron en su matriz de confusión no fueron balanceados o iguales entre los datos positivos o negativos para su evaluación, en cambio en nuestra matriz sí, esto puede verse en la Figura 4. Por lo cual, ese 98.17% es un porcentaje dudoso y no es seguro que sea válido totalmente. Además, que la cantidad de imágenes para entrenar a la red neuronal implica un trabajo más prolongado, mientras que nuestro software es más accesible a implementar en un tiempo corto

Conclusiones

Se desarrolló satisfactoriamente con un porcentaje de precisión alto un algoritmo de visión artificial, ver la Figura 3, para identificar los defectos físicos de la lata de atún redonda, este sistema es muy útil para el control de calidad de las industrias que trabajan con este objeto. En especial para nuestro país, en donde en diferentes lugares aún siguen haciendo su control de calidad con supervisión del mismo personal. Se implementó un ambiente controlado en el proyecto, ver la Figura 1, ya que es necesario hacer las pruebas en un ambiente que se adapte a las condiciones con las que estamos haciendo la programación. Si es que hacemos las pruebas en un lugar más iluminado que otro, entonces el sistema no va a responder de la misma manera. Por lo que, para implementarlo dentro de las empresas debe ser con una especial medición y evaluación de las condiciones o el ambiente en el cual va a actuar este sistema, porque sabemos que no todas las empresas tienen las mismas condiciones.

Se aplicaron las técnicas y sus funciones principales pueden verse en la Tabla 1-5, como la escala de grises, borde de imagen, filtro gaussiano y operación morfológica, y pudimos concretar que se puede realizar un buen diagnóstico, a través del sistema propuesto. Concluimos que, los resultados de nuestro proyecto fueron adecuados y precisos en un 96.08% y puede identificar las latas buenas y los distintos golpes que suelen tener las latas de atún redondas en las industrias con precisión. En tal sentido, es muy importante que este tipo de sistemas comiencen a implementarse en las empresas para poder mejorar en términos de calidad, cuidado al consumo humano y reducción de costos.

Referencias

- Aguilar, M. J. (2017). Escala de grises. *Glosario C+R*. Universidad Politecnica de Valencia, 2017. <http://glosario.ldr.webs.upv.es/postout/3716/escala-de-grises>.
- Alvarado, C. A. (2021). Implementación de la 5S para mejorar la gestión del almacén de una institución policial, La Libertad 2020 [Tesis de licenciatura, Universidad Privada del Norte]. Repositorio de la Universidad Privada del Norte. <https://hdl.handle.net/11537/29032>
- Beuzen, T., & Timbers, T. (2022). *Python Packages*. Chapman and Hall/CRC.
- Caravedo, B. (2016). La energía social en las empresas B. *Universidad del Pacífico, Lima*.
- Cardellino, F. (2021). La guía definitiva del paquete NumPy para computación científica en Python. *freeCodeCamp.org*. [La guía definitiva del paquete NumPy para computación científica en Python](https://www.freecodecamp.org/es/learn/python/computacion-cientifica-en-python/)
- Challenger-Pérez, I., Díaz-Ricardo, Y., & Becerra-García, R. A. (2014). El lenguaje de programación Python. *Ciencias Holguín*, 20(2), 1-13. <https://www.redalyc.org/pdf/1815/181531232001.pdf>
- Diario de Chimbote (2020). *Incautaron 40 cajas de conserva de dudosa procedencia ayer en el mercado La Perla*. [Noticia] Chimbote. <https://diariodechimbote.com/>
- Diaz, D. (2023). Guía para principiantes de la programación orientada a objetos (POO) en Python. *Kinsta*. [En línea] 15 de Junio de 2023. [Citado el: 03 de Noviembre de 2023.] <https://kinsta.com/es/blog/programacion-orientada-objetos-python/>.

- Dominguez M., T. (2021). *Visión artificial. Aplicaciones prácticas con Open CV-Python*. Primera. Barcelona : Marcombo.
- i García, E. M. (2012). Visión artificial. *Fundación para la universidad Oberta de Catalunya*. <https://www.playmodes.com/wp-content/uploads/2017/04/UOC-PDF-6.pdf>
- Mehmood, S. (2022). OpenCV Dilate. *Delftstack*. [Citado el: 3 de enero de 2023.] <https://www.delftstack.com/es/howto/python/opencv-dilate/>.
- Mendoza, A. H. (2023). *Gestión de la innovación tecnológica: del aula a las soluciones*. Universidad Iberoamericana AC.
- Munoz, Niña (2023). Defectos de las latas: causas, efectos y prevención. *INNOSEN*. [En línea] Enero de 2023. <https://www.innosen.com/es/articulos/defectos-de-las-latas-causas-efectos-y-prevencion-2/>.
- Ortiz R., E., Mejía-Lavalle, M., & Sossa, H. (2017). Filtrado de ruido Gaussiano mediante redes neuronales pulso-acopladas. *Computación y Sistemas*, 21(2), 381-395. <https://doi.org/10.13053/cys-21-2-2742>
- Otarola, Valeria (2014). Parámetros de calidad en el atún enlatado. *Prezi*. [En línea] 2014. https://prezi.com/m9_kioa3rwe1/parametros-de-calidad-en-el-atun-enlatado/.
- Palma S., W. S. (2020). Implementación de un sistema de visión artificial en líneas de producción de atún, para la detección de fallas de codificado en la empresa Puertomar SA, Provincia de Manabí. <http://dspace.esPOCH.edu.ec/handle/123456789/14101>
- Palma S., W. S. (2020). Implementación de un sistema de visión artificial en líneas de producción de atún, para la detección de fallas de codificado en la empresa Puertomar SA, Provincia de Manabí. <http://dspace.esPOCH.edu.ec/handle/123456789/14101>
- Poblet G., P. (2022). *Detección de defectos en latas de refrescos mediante redes neuronales y visión artificial* (Bachelor's thesis, Universitat Politècnica de Catalunya). <http://hdl.handle.net/2117/371689>
- Rebaza, J. V. (2007). Detección de bordes mediante el algoritmo de Canny. *Escuela Académico Profesional di Informática. Universidad Nacional de Trujillo*, 4.
- Rico-Schmidt, E. (2019). OS – Acceso portable a funciones específicas del sistema operativo. <https://rico-schmidt.name/pymotw-3/os/index.html>
- Rodríguez, H. (2021). ¿Qué es OpenCV? ¡Descubre todo acerca de la visión artificial! *Crehana*. <https://www.crehana.com/blog/transformacion-digital/que-es-opencv/>.
- Romero, J. G. I. G. (2022). *Investigación organizacional: Desafíos y perspectivas*. Universidad Juárez del Estado de Durango.
- Rosebrock, A. (2015). I just open sourced my personal imutils package: A series of OpenCV convenience functions. *Pyimagesearch*. [My imutils package: A series of OpenCV convenience functions - PyImageSearch](https://pyimagesearch.com/2015/02/16/i-just-open-sourced-my-personal-imutils-package-a-series-of-opencv-convenience-functions-pyimagesearch/)
- Sanchez, D. (2023). *Indecopi sanciona a 6 supermercados por vender productos enlatados con golpes y abolladuras*. [Noticia], Perú : La Republica. <https://larepublica.pe/>
- Severance, C. R., Castellanos, J. C. P., & de Dios Dougnac, J. (2009). *Python para todos: explorando datos en Python 3*. Charles Severance.
- Soria, E., Rodríguez, P., García, Q., Vaquer, F., Vicent, J., & Vila, J. (2022). *Inteligencia artificial*. Ra-Ma Editorial.