

# DEVELOPMENT QUALITY CONTROL SYSTEM THROUGH ARTIFICIAL VISION FOR THE DETECTION OF STAINS GENERATED BY CELL BREAKS IN CANNED ARTICHOKE

Ryan Abraham León León<sup>1</sup> ; Irvin Eduardo Alvarado Avalos<sup>1</sup> 

<sup>1</sup>Universidad Privada del Norte, Perú, ryan.leon@upn.edu.pe; <sup>1</sup>Universidad Privada del Norte, Perú, N00240502@upn.pe

*Abstract – This research work put into practice new technologies related to artificial vision so that, through the use of neural networks, specifically convolutional ones, the general objective can be achieved by developing a quality control system through artificial vision. In order to detect the different stains generated by cellular breaks that may exist in preserved artichokes. In order to execute artificial vision, the project was supported by the Python program, which made it easier to handle the programming language together with PyCharm, both of which are an ideal complement so that correct coding and development of the desired system can be developed. Furthermore, artificial vision is considered one of the most efficient methods, since, in its development, it works with a large visual database, which, in the training stage, is allowed to learn and subsequently be able to predict and recognize on its own what you are showing it, generating the desired detection. However, this use is linked to constant research so that it can work correctly. Finally, after making a precision table, it showed that the system manages with 98% of this, compared to other investigations that manage percentages of 83.9% and 96.6%. This allows us to conclude that the work carried out based on a large collection of data works correctly, allowing its future implementation in the quality area.*

*Keywords---* Artificial vision, Artichoke, Quality, Detection, Python.

**Digital Object Identifier:** (only for full papers, inserted by LACCEI).  
**ISSN, ISBN:** (to be inserted by LACCEI).  
**DO NOT REMOVE**

# DESARROLLO DE UN SISTEMA DE CONTROL DE CALIDAD A TRAVÉS DE LA VISIÓN ARTIFICIAL PARA LA DETECCIÓN DE MANCHAS GENERADAS POR ROTURAS CELULARES EN LAS CONSERVAS DE ALCACHOFA

Ryan Abraham León León<sup>1</sup> ; Irvin Eduardo Alvarado Avalos<sup>1</sup> 

<sup>1</sup>Universidad Privada del Norte, Perú, ryan.leon@upn.edu.pe; <sup>1</sup>Universidad Privada del Norte, Perú, N00240502@upn.pe

**Resumen**– El presente trabajo de investigación puso en práctica las nuevas tecnologías relacionadas con la visión artificial para que, mediante el uso de redes neuronales, específicamente las convolucionales, se pueda conseguir el objetivo general de desarrollar un sistema de control de calidad mediante la visión artificial para poder detectar las diferentes manchas generadas por roturas celulares que puedan existir en las conservas de alcachofa. Para poder ejecutar la visión artificial, el proyecto se apoyó en el programa Python, el cual permitió manejar con mayor facilidad el lenguaje de programación junto a PyCharm, ambos son un complemento ideal para que se pueda desarrollar una correcta codificación y una elaboración del sistema deseado. Además, la visión artificial se considera como uno de los métodos más eficientes, ya que, en su desarrollo, este trabaja con una gran base de datos visuales, la cual, en la etapa de entrenamiento, se le permite aprender y posteriormente poder predecir y reconocer por sí solo lo que le estás mostrando, generando la detección deseada. Sin embargo, este uso se ve ligado a una investigación constante para que pueda funcionar de manera correcta. Finalmente, tras realizar una tabla de precisión, arrojó que el sistema se maneja con un 98% de esta, a comparación de otras investigaciones que manejan porcentajes de 83.9% y 96.6%. Esto permite concluir que el trabajo realizado en base a una gran recolección de datos funciona correctamente, permitiendo su futura implementación en el área de calidad.

**Palabras Clave**-- Visión artificial, Alcachofa, Calidad, Detección, Python.

## I. INTRODUCCIÓN

El Perú es un país donde hay bastante demanda de alimentos procesados, estos en su mayoría se manejan en material de conservas, ya que se busca su preservación por un tiempo mayor al normal [1]. Las conservas pasan por un control de calidad, ya que gracias a ello se identifican los productos que están defectuosos y productos buenos. También los controles de calidad permiten aplicar sistemas de mejoras para poder lograr productos de mayor calidad y satisfacción de los clientes. A

diferencia de otros sectores alimentarios, el proceso de producción de la industria de conservas tiene más holgura al producir sus productos, porque se tiene un bajo riesgo microbiológico; sin embargo, nunca se debe descartar la posibilidad de riesgo. Esto se demuestra en el año 2022, en el cual se exportaron 58.319 toneladas de alcachofa y generó 162 millones. Este resultado se evidenció en un aumento del 5% en cantidad y del 8% en términos monetarios. Al mismo tiempo, el precio medio obtenido por los exportadores peruanos experimentó un incremento del 3%, alcanzando un promedio de USD 2.77 por kilogramo [2]. La alcachofa se posicionó como la segunda hortaliza más destacada en las exportaciones de Perú. Este producto llegó a más de 20 países en cuatro continentes y la principal presentación en la se exportó fue como conserva procesada. Es así como en la actualidad la calidad de las conservas es muy importante ya que su demanda ha aumentado gracias a sus consumidores, se ha dicho que las conservas son consideradas más apropiadas al estilo de vida rápida. No obstante, estas son clasificadas como estériles, pero en muchas instancias, existe la posibilidad de que se puedan llegar a contaminar con microorganismos patógenos resistentes al calor, tanto antes como después del tratamiento térmico [3]. Por esta razón, se propuso la implementación de un sistema de aseguramiento de la calidad. en base a una herramienta llamada Python la cual sirve para la detección de las manchas en las conservas.

Python es un lenguaje de programación que ha tenido un gran crecimiento y se aplica al mundo real. Este lenguaje es dinámico, flexible, ofrece orientación de objetos, con aplicación en la informática. Tiene un diseño sencillo y fácil de entender. Utiliza diferentes formas de programación, el cual lo convierte en un lenguaje que puede ser utilizado por principiantes y reemplaza a algunos lenguajes como Java [4]. Además, este lenguaje tiene disponibilidad gratuita y los que desarrollan este lenguaje pueden fácilmente leer y traducir el código Python, en comparación de otros tipos de lenguajes. Así que, tiene menor costo para mantenerlo y desarrollarlo, sin ninguna barrera. Al desarrollarlo, se puede utilizar en otros

**Digital Object Identifier:** (only for full papers, inserted by LACCEI).  
**ISSN, ISBN:** (to be inserted by LACCEI).  
**DO NOT REMOVE**

proyectos, ya que se puede importar o exportar [5]. La alcachofa (*Cynara Cardunculus* var. *scolymus*) es una hortaliza de alto valor comercial en muchas partes del mundo debido a su sabor único y beneficios nutricionales [6]. Sin embargo, el cultivo de alcachofa puede estar sujeto a muchos problemas y pérdidas de producción que pueden afectar negativamente al rendimiento y la calidad. Un factor que afecta significativamente a la pérdida de producción en el cultivo de alcachofa es el medio ambiente. Los elementos climáticos, como la temperatura, la humedad y la accesibilidad al agua, son elementos cruciales que influyen en el crecimiento y desarrollo de las plantas de alcachofa. [7]. Las enfermedades y plagas también son factores importantes en las pérdidas de producción de alcachofa. Los patógenos fúngicos, bacterianos y virales pueden afectar las plantas y reducir el rendimiento [8]. Además, insectos como el pulgón negro de la alcachofa (*Brachycaudus cardui*) pueden causar daños importantes. Unas prácticas de gestión inadecuadas pueden provocar una pérdida de rendimiento en el cultivo de alcachofa. Esto incluye la selección de cultivares adecuados, fertilización, riego y técnicas de poda [9]. Una mala gestión también puede provocar la propagación de enfermedades y plagas. El uso de tecnología avanzada, como sistemas de riego eficientes y el uso de invernaderos, ayuda a minimizar las pérdidas de producción causadas por el medio ambiente [10]. Estas tecnologías nos dan un mejor control sobre las condiciones de crecimiento.

A partir de todo lo presentado, hay trabajos de investigación relacionados con la detección de algún tipo de mancha, suciedad y defecto en los diferentes productos que se comercializan. Es así como en la investigación [11], en donde se tiene como objetivo general el desarrollar un algoritmo para la identificación de suciedad o manchas presentes en la cáscara del huevo de gallina, utilizando técnicas de visión artificial. Para lograr este objetivo, se adquirió un total de 270 imágenes de tres cubetas de huevos, las cuales fueron clasificadas en sus diferentes presentaciones comerciales. Se hizo uso de una cámara CCD para la captura de imagen y poder transformarlo a RGB en formato de JPG, estas fueron transformadas al espacio de color YCrCb, en donde se empleó el método de Otsu, extrayendo características como el área, perímetro, redondez y los radios de la elipse descrita por el huevo, asimismo el área de suciedad. Es así como para el proceso, se usó el método de regresión múltiple y el método Forge, logrando desarrollar el algoritmo deseado, el que, tras puesta a pruebas, brindó un porcentaje de acierto del 74.71%. Por otro lado, también existen numerables investigaciones donde se aplica la visión artificial para la mejora de la calidad de sus productos, como en una investigación [12], donde se plantea como objetivo el mejorar su control de calidad de los productos que ofrecen, es por ello por lo que se necesita una automatización de procesos para evitar pérdidas financieras por pérdidas de tiempo y errores de fabricación. El diseño tiene una estructura metálica y una bandeja especial con 36 huevos pollo colocándolos en un panel LED para iluminación. Utiliza un mecanismo de transferencia bidimensional basado en impresoras 3D. Hay fotografías

procesadas con diferentes indicadores gráficos para controlar la calidad, es así como, durante cada análisis, los resultados se registran en la base de datos, y se generan datos e informes correspondientes. El proceso de imágenes tiene una duración de 61,92 segundos que, si se desea obtener el dato por huevo, sería un tiempo de 1,72 segundos por cada uno. El diseño fue puesto a prueba con huevos para consumo humano resultando en 108. Estos se definieron como problemáticos, 21% aptos y 79% no aptos. Los principales problemas fueron: presencia de grietas, una alta porosidad y cáscaras delgadas. El diseño detecta problemas con la estructura del huevo, sin embargo, no se puede comprobar que los 100 seleccionados como aptos proceden de gallinas, 29 por las condiciones biológicas, el tratamiento y la incubadora también tienen su influencia al usar el sistema de procesamiento de imágenes que limita la eficiencia del procesamiento de iluminación y cámara digital. Por otro lado, también se tiene un trabajo [13], el cual tuvo como objetivo principal crear un sistema, una visión artificial que permite una clasificación de los limones según sus formas y códigos límite. Investigar y encontrar la solución que fue diseñada según las etapas de la visión artificial. (adquisición, preprocesamiento, separación, especificación, detección e interpretación), donde se compararon los sistemas de cada paso para determinar cuáles son los más apropiados para el problema. Después de haber desarrollado el sistema de visión artificial utilizando el algoritmo seleccionado, se encontró que la precisión del programa era de un 83.9%. En cuanto al tiempo de procesamiento, se logró clasificar los limones en tan solo 0,33 segundos. Estos resultados confirmaron la hipótesis de que un sistema de visión por computadora es efectivo para la categorización de limones. En otra tesis investigada [14], el objetivo principal fue crear un algoritmo de detección de frutos fuera de forma mejorar la calidad; para esto, se procesó imágenes digitales, incluyendo análisis de histogramas, aplicación de operadores de morfología y la conversión a otros espacios de color. En el estudio se utilizó MATLAB, en donde se tuvo 31 imágenes de entrada simuladas que se basaron en métodos que determinaron que los frutos mediante el análisis hecho dieran como resultado si tuvo un buen comportamiento o no, tratando de determinar qué tipo de falla se identifica. En el capítulo 2, se abarcaron las diferentes técnicas utilizadas con relación a las imágenes. En el capítulo 3, la imagen se convierte a una matriz de formato doble, luego la imagen segmentada se transmitió al sistema. Se buscó obtener el valor resuelto y así determinar si es nulo el error de resultados y observaciones obtenidos. Posteriormente, enviaron 100 imágenes escaneadas, 11% fallaron debido a este tipo de clasificación errónea y el 89% acertó donde el resultado indicaba una tasa de éxito favorable. Finalmente, se ha logrado aplicar con éxito los algoritmos, por ejemplo: detección de elementos extraños a través de la segmentación, análisis de bordes y el estudio del histograma con un tiempo de respuesta de 69,5 segundos por cada una. En la investigación [15], se plantea el objetivo de desarrollar una red neuronal convolucional que permita la identificación de la madurez adecuada del plátano. En Python,

se utilizó la librería GPU, en el que se ha configurado un entorno virtual para poder utilizar la GPU de la laptop que facilita el entrenamiento de una red neuronal. Se utilizaron 480 imágenes, con ello se logró una precisión del 98.10%, una sensibilidad del 85.89% y un accuracy del 92.08%. Asimismo, en un estudio [16], se plantea el objetivo de desarrollar visión artificial para reconocer el estado de madurez de granadillas a partir de la detección de imágenes. En Python, utilizaron la librería OpenCV para la extracción de los pixeles que tiene una imagen, luego se clasificaron por medio del agrupamiento, con 110 puntos RGB, que pertenecían al estado de madurez de la granadilla. A partir de imágenes de 90 frutas en distintos niveles de maduración, se tuvo como resultado 93 % de exactitud y 96,6 % de precisión en las frutas maduras y verdes.

Es por lo que el trabajo que se realiza en la siguiente investigación ya tiene una base que ha sido recolectada a lo largo de los años y ahora mediante la inteligencia artificial se puede complementar y mejorar el sistema de visualización y detección. Es así como en el presente estudio se tiene como objetivo general desarrollar un sistema de control de calidad mediante la visión artificial para poder detectar las diferentes manchas que puedan existir en las conservas de alcachofa. Por otro lado, los objetivos específicos vienen a ser la investigación del programa Python, acompañado del modelo de la red neuronal convolucional que en este caso viene a ser la red YOLOv5. También la recolección del material visual necesario, con la finalidad de crear una base de datos para que funcione correctamente la detección. Finalmente, la creación de la codificación correcta y la aplicación de la red necesaria para el desarrollo exitoso del sistema de calidad.

## II. MÉTODOS Y MATERIALES

### A. Las Redes Neuronales

Para el desarrollo del sistema, se tuvo que estudiar el mundo de las redes neuronales, para que se pueda identificar qué tipo se adecuaba más al objetivo propuesto y aplicarlo en el presente trabajo. Estas redes fueron en principio una simulación de los sistemas nerviosos biológicos; estas fueron formadas por un conjunto de cuerpos que son comúnmente conocidos como neuronas. La mayoría de las neuronas presentan una estructura en forma de árbol llamada dendritas, que reciben señales de entrada de otras neuronas a través de sinapsis, mientras que el axón se encarga de transmitir las señales de salida de las neuronas. Esto es fundamental para cumplir el objetivo principal de estas redes, que consiste en llevar a cabo operaciones de síntesis y procesamiento de información.

Por otro lado, están las famosas Redes Neuronales Artificiales (ANN), las cuales se basan en el funcionamiento de las redes biológicas, pero presentan otras funcionalidades y configuraciones de conexiones que difieren de las observadas en el contexto biológico. Como características principales se tiene en primer lugar, a la Autoorganización y Adaptabilidad,

en las que se emplean algoritmos de aprendizaje que proporcionan mayores oportunidades para un procesamiento resistente y adaptable. En segundo lugar, se encuentra el Procesamiento no Lineal, este enfoque se fundamenta en la mejora de la red para aproximarse a funciones y clasificar patrones, al mismo tiempo que fortalece su capacidad de resistencia al ruido. Finalmente, el procesamiento paralelo, el cual consiste en dividir una tarea informática en subtarear que pueden ser ejecutadas simultáneamente por distintos procesadores, con el fin de acelerar la finalización de la tarea en su conjunto y reducir el tiempo de ejecución.

El componente fundamental de la computación en este contexto, que se asemeja a una neurona en un modelo de red, esta suele denominarse nodo o unidad. Este nodo recibe una entrada desde otras unidades o desde una fuente externa. A cada ingreso se le asigna un peso, representado como "w", y estos se ajustan durante el entrenamiento. Cada unidad realiza una operación específica, aplicando la función que relaciona la suma de los inputs ponderados por sus respectivos pesos, como se puede ver en la Fig. 1.

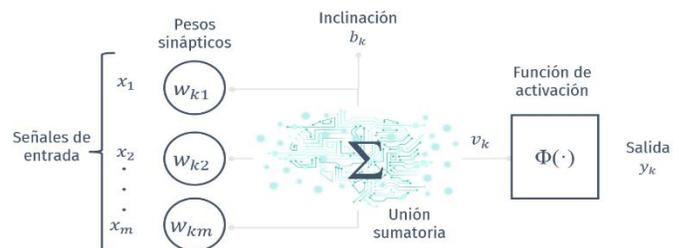


Fig. 1 Modelo genérico de una neurona artificial.

Los componentes de las ANN desempeñan un trabajo crucial en aplicaciones relacionadas al uso para procesar imágenes o señales. En estos casos, se utilizan arquitecturas que incluyen elementos de procesamiento adaptativo y paralelo, combinados con estructuras de interconexión jerárquica [17].

### B. Función de activación

Una función de activación es una función que se utiliza para transmitir la información que proviene de la suma de los pesos y las entradas en una red neuronal. En otras palabras, estas funciones son el medio a través del cual se comunica la información a través de las conexiones de salida. Además, facilita el entrenamiento evitando la saturación de la neurona y así logra mejorar el rendimiento de la red.

La función Leaky ReLU (Rectified Linear Unit con fuga) que se muestra en (1) es una variante de la función ReLU (Rectified Linear Unit) que se utiliza comúnmente en redes neuronales artificiales.

$$f(x) = \begin{cases} 0, & \text{for } x < 0 \\ a * x, & \text{for } x \geq 0 \end{cases} \dots (1)$$

### C. Red Neuronal Convolutiva (CNN)

Es así, como de las redes con más utilización en la actualidad es la convolutiva, las CNN son un tipo de red diseñada particularmente para el procesamiento estructurado en forma de malla, como imágenes y videos. Estas emplean una operación matemática denominada convolución para identificar características significativas en los datos de entrada y aprender a reconocer patrones en ellos. Una CNN combina varias capas conectadas que están destinadas a la inspección de forma automática de características, dirigidas a la clasificación [18]. Por ello, se destacan en aplicaciones de visión por computadora, como la agrupación de imágenes, la detección de objetos y la segmentación semántica. La habilidad de aprender y extraer automáticamente características relevantes ha tenido un importante impacto en el mundo de la visión por computadora y ha posibilitado la creación de aplicaciones prácticas, como la detección de rostros, la categorización de imágenes y la conducción autónoma en vehículos [19].

La arquitectura CNN de la Fig. 2 se compone de 24 capas convolucionales, 4 de agrupación y 2 totalmente conectadas, en las primeras se identifican las características de la imagen, mientras que las capas totalmente conectadas se encargan de prever los porcentajes de salida y su ubicación en la captura. La información de entrada consiste en una imagen de  $448 \times 448$  y 3 canales de color. La información de salida se representa mediante un tensor de dimensiones  $7 \times 7 \times 30$ . Este tensor consta de  $7 \times 7$  casillas, cada una equipada con 30 neuronas. De las cuales, 10 se dedican a las coordenadas de las dos ventanas, mientras que las restantes 20 se utilizan para las puntuaciones correspondientes a cada una de las clases [20].

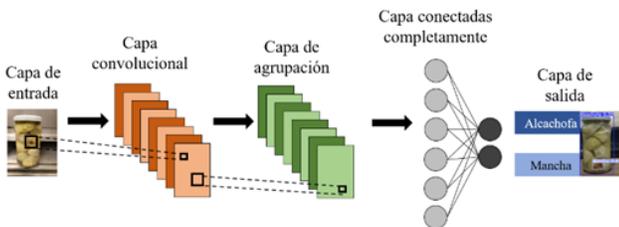


Fig. 2 Arquitectura de la CNN Yolo.

Las redes neuronales convolucionales están compuestas de tres tipos principales de capas.

### D. Capa Convolutiva

La capa convolutiva constituye la parte fundamental de una CNN, siendo el lugar principal donde se efectúan la mayoría de los cálculos. Para su funcionamiento, necesita elementos como la información de entrada, los filtros a aplicar y un mapa de características. También se pueden utilizar fuentes en línea para realizar las convoluciones, como en este caso que se usó Google Colaboratory que se puede observar en la Fig. 3, la cual se encargó de tomar la imagen original, analizarla mediante la convolución y obtener ciertas características que se tienen en esa imagen para posteriormente redactarlo en otra matriz, es decir, es así como se mueve por los

campos receptivos de la imagen para verificar la existencia de la característica [21].



Fig. 3 Uso del filtro escala de grises.

El detector de características consiste en una matriz de pesos 2D que representa una sección de la imagen. Los filtros se emplean en un área de la imagen y luego se realiza una operación de producto escalar entre los píxeles de entrada y el filtro correspondiente, cuyo resultado se agrega a una matriz de salida. Dando como resultado final, un mapa de características, mapa de activación o característica convolucionada.

### E. Capa de Agrupación

La agrupación de capas, también conocida como submuestreo es la que logra que se genere una reducción de la dimensionalidad al disminuir el número de parámetros de entrada. Durante esta operación, se aplica un filtro que barre toda la entrada. Luego se inserta una función que altera los datos dentro del campo receptor y llena así la matriz de salida. Esta capa ayuda a disminuir la complejidad y hacer más efectiva la ejecución. Los dos tipos principales de agrupación son la máxima y la media.

### F. Capa Totalmente Conectada (CTC)

El nombre de la capa totalmente conectada describe con precisión la capa en sí. En la CTC, cada nodo de salida está directamente vinculado a un nodo de la capa previa. Esta capa lleva a cabo la tarea de clasificación utilizando las características extraídas de las capas anteriores y sus diversos filtros. [22].

### G. Estructura de la CNN

La capa de entrada de la imagen es una fase de identificación de las imágenes de entrada. En esta capa las imágenes se redimensionan, rotan e incluso pueden tomar pequeñas muestras al azar [23].

Por otro lado, la capa convolutiva trabaja con las imágenes de entrada para poder obtener el mapeo de las características.

A su vez, se tiene la capa de agrupamiento, Esta capa realiza un submuestreo para reducir el tamaño espacial de las capas convolucionales y lograr cierta invariancia traslacional. Una forma frecuentemente empleada es la agrupación máxima, utilizando regiones de tamaño  $2 \times 2$  y un paso de 2.

También, está la CTC, en donde la salida de la última capa convolucional o de agrupación se dirige hacia una o CTC, de manera análoga a una red neuronal convencional. Las características extraídas en las capas anteriores son empleadas para llevar a cabo la clasificación.

Una de las últimas capas, es la de clasificación a través de SoftMax, la red varía en la precisión con las que las neuronas aprenden debido a que se inicializan los pesos aleatoriamente. Después, se aplica una función de pérdida de entropía cruzada conocida, expresada en la ecuación (2). Esta función se utiliza para calcular la discrepancia entre la etiqueta del valor real y la etiqueta predicha.

$$Softmax(Y_i) = \frac{e^{Y_i}}{\sum_{j=1}^n e^{Y_j}} \dots \quad (2)$$

Donde  $y_i$  representa a la cantidad de salidas de la red neuronal y la suma abarca desde la primera clase hasta la última desarrollada que, en este trabajo, se aplicaron solo dos, la etiqueta de “Alcachofa” y la de “Mancha”, que viene a ser la cantidad de clases en la red neuronal. La salida de esta capa indica la probabilidad de que la imagen pertenezca a cada una de las etiquetas.

Finalmente, se encuentra la capa de salida, esta es la última que se encarga de mostrar el porcentaje de éxito de clasificación.

#### H. Aplicación de la CNN

Las redes neuronales convolucionales son un algoritmo de Deep Learning o Machine Learning que están diseñadas para trabajar con imágenes. Este algoritmo ha sido fundamental en el avance del campo de Visión por Computadora. [24]. Las redes neuronales convolucionales son muy potentes para todo lo que tiene que ver con el análisis de imágenes, debido a que son capaces de detectar características simples como por ejemplo la detección de bordes, líneas, etc. y componer en características más complejas hasta detectar lo que se busca [25]. Esto se logra en el proceso de aprendizaje, donde entrenar una red significa ajustar los parámetros, pesos y umbrales de la red para que la entrada presentada produzca la salida deseada, es decir, buscar no tener errores de predicción. La selección del diseño óptimo en este trabajo se determinó utilizando medidas de evaluación de pronóstico aleatorias y no aleatorias tradicionales [26]. Aplicar las redes convolucionales es considerada como un avance que debe ser aprovechado, ya que se puede orientar a la identificación de patrones en la agricultura de precisión, como lo fue en su investigación del estudio del tizón tardío en el cultivo de papa [27].

#### I. Materiales

Después de conocer el comportamiento y funcionamiento de la red que se plantea, se procede a detallar los materiales necesarios para el desarrollo del sistema, a través de una codificación que se apoya en la visión artificial.

Con respecto a la recolección de material visual, se necesitan cientos de imágenes para el entrenamiento y validación de estas, para ello, se debe tener fotografías en donde se pueda observar la conserva de alcachofa con manchas que pueden tener una vez terminado el proceso de envasado, además también se requieren fotos de la conserva de alcachofa en perfecto estado, que le permitirá al algoritmo diferenciar entre una conserva en buen estado o una que cuenta con algún tipo de mancha Cabe recalcar, que todas las imágenes extraídas han sido tomadas dentro de la planta procesadora de alcachofa, mostrando el problema existente en las empresas con respecto al desarrollo de conservas.

Por otro lado, es necesario el manejo de una cámara o de un teléfono celular que cuente con una alta resolución para poder utilizarla en el sistema, de modo que se encargará de capturar y transmitir las imágenes en tiempo real hacia el algoritmo, dando pase a la ejecución de la codificación programada para la identificación de la conserva en buen estado o con mancha.

También el uso de una laptop que cuente con una capacidad de procesamiento idónea, ya que será la base del funcionamiento del programa Python, el cual se apoyará en PyCharm, utilizando diferentes librerías para completar la codificación. En la Tabla I se describe a mayor profundidad lo necesario para el desarrollo de la codificación.

TABLA I  
COMPONENTES PARA EL DESARROLLO DE LA CODIFICACIÓN

Componente	Especificación
Procesador	11th Gen Intel(R) Core (TM) i7-1165G7 @ 2.80GHz 2.80 GHz
RAM instalada	16.0 GB
Tipo de sistema	Sistema operativo de 64 bits, procesador x64
Edición	Windows 11 Pro
Python	Versión 3.11
PyCharm	Versión 2023.2.1
Librerías	OpenCV
	Numpy
	PyTorch

#### J. Etiquetado de imágenes

Una vez que se recolectaron las imágenes suficientes para empezar a trabajar el algoritmo, se separan en dos lotes, un porcentaje se dirige a la carpeta llamada train y la otra a val, con el objetivo de que los etiquetados se guarden en sus carpetas correspondientes relacionadas con sus respectivas imágenes. Las etiquetas se conformaron por dos, una denominada “alcachofa” y otra que se denominaba “mancha” Para el

etiquetado se utilizó la página Make Sense como se puede ver en la Fig.4.

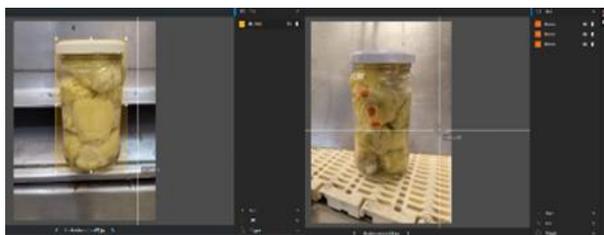


Fig.4 Etiquetado de imágenes.

### K. Entrenamiento

En la etapa de entrenamiento, la cual es la más importante en todo el desarrollo del sistema, se usó de apoyo Google Colaboratory con YOLOv5 que cuenta con una arquitectura como se observa en la Fig. 2, ahí es donde se codificó usando la nube buscando que el modelo se entrene correctamente como se muestra en la Fig. 5, este entrenamiento consta de la entrada de las 1054 imágenes que anteriormente fueron etiquetadas como se observa en la Fig. 4, estas ingresan con sus pesos sinápticos como se señala en la Fig. 1, en donde conforme se vayan entrenando, se van modificando los pesos de manera de que sean más precisos, estos se ajustan de acuerdo a las redes convolucionales que aplican el filtro que se indica en la Fig. 3 y así se desarrolló el entrenamiento en base a 55 épocas.

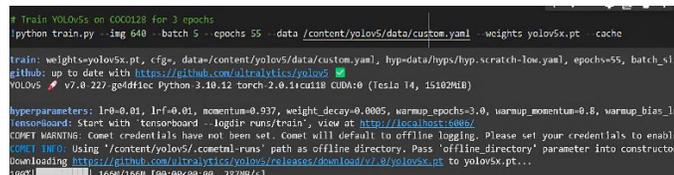


Fig. 5 Entrenamiento del modelo

### L. Codificación

En primer lugar, se inició la codificación, importando las librerías necesarias para el desarrollo del algoritmo.

Importación de las librerías al programa	Descripción
import torch	Evento en el que se importan las librerías a utilizar para la codificación
import cv2	
import numpy as np	
import pandas	

Fuente: Autor

Se hace el llamado a las librerías para el funcionamiento, estas vienen a ser la librería torch que es utilizada para la aplicación del conocido método Maching Learning con el reconocimiento de imágenes y aprendizaje profundo. Por otro lado, el llamado de cv2 permite el trabajo con imágenes en el programa Python. Finalmente, la librería numpy que para hacer más fácil su llamado se abrevió en np, se encarga de realizar cálculos lógicos matemáticos sobre cuadros y matrices. Finalmente, la librería panda, la cual brinda los datos de la detección como coordenadas, precisión, etc.

Tras la importación de librerías, se procede a leer el modelo, aquí es donde trabaja todo el algoritmo, ya que previamente se han etiquetado las imágenes, estas se entrenaron correctamente con YoloV5 y se convirtió el modelo en un archivo compatible con Python.

Lectura del modelo	Descripción
model = torch.hub.load('ultralytics/yolov5', 'custom',	Evento en el que se importa el modelo previamente trabajado, para su correcta lectura en el programa
path = 'C:/Users/irvin/OneDrive/Escritorio/ Python/Video/model/Alcachofa.pt')	

Fuente: Auto

El objetivo de este código es el cálculo y extracción de todo lo almacenado previamente en el modelo, es decir, los pesos correspondientes de cada etiqueta y el entrenamiento que se encargó de relacionar y reconocer imagen-etiqueta.

Finalmente, programamos el funcionamiento de la captura de imagen, indicando la cámara de vídeo encargada para el funcionamiento del algoritmo.

Activación de cámara y modelo	Descripción
cap =cv2.VideoCapture('http://192.168.1.5: 8080/video') while True: ret, frame = cap.read() detect = model(frame) cv2.imshow('Detector de Manchas', np.squeeze(detect.render())) t = cv2.waitKey(5) if t == 27: break	Evento en donde se configura la cámara a utilizar para la ejecución, por otro lado, activa la detección a través del modelo, Es ahí donde se hace presente la función Leaky ReLu como se muestra en (1). Finalizando, se crea una salida en el teclado que viene a ser "Esc", para poder detener la ejecución.

Fuente: Autor

Tras finalizar la codificación del sistema que se buscaba desarrollar inicialmente, se muestra en la Fig. 6 un resumen de los pasos que se siguieron para la elaboración exitosa del sistema de control de calidad.

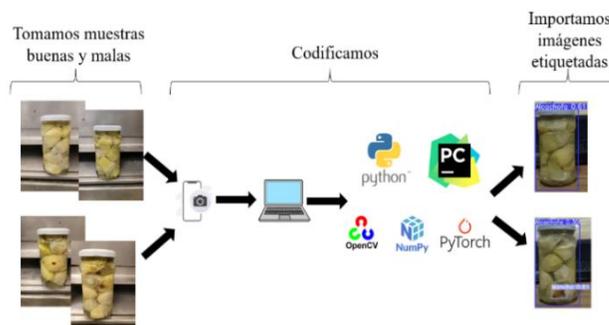


Fig. 6 Vista general del desarrollo del sistema de calidad

### III. RESULTADOS

Una vez realizado el sistema de calidad, se muestran los siguientes resultados que da el funcionamiento de la programación mediante visión artificial, como se puede observar en la Fig. 7, en donde permite ver la detección en tiempo real de las diversas manchas que puedan tener las conservas, para caso de prueba se colocó una conserva en perfecto estado y otra con las manchas, esto se hizo con la finalidad de probar que la red neuronal funciona correctamente, ya que la función de activación (1) está arrojando los resultados deseados. Asimismo, este sistema de calidad puede detectar diversas manchas a la vez de diferentes conservas, permitiendo un aprovechamiento máximo al momento de su implementación en planta.



Fig. 7 Captura de video en tiempo real

Por otro lado, analizando la Fig. 8, se puede llegar a ver un xmin, ymin, xmax y ymax, que hacen referencia a la posición de la mancha en coordenadas con respecto a la imagen capturada, estos datos se muestran para cada mancha detectada. Además, también indica el porcentaje de seguridad, es decir, si indica un 0.652597, está haciendo referencia a que, según el programa, está seguro de que lo detectado es una mancha al 65%, y así consecutivamente con el resto que pueden llegar a ser detectadas. Este porcentaje es resultado de la ecuación (2), ya que relaciona los pesos con las etiquetas que en este caso se manejan dos, las cuales se denominan “Alcahofa” y “Mancha”.

```

Run  Detector de manchas final x
┌───┴───┐
│ 1 1100.619873 729.672363 1198.203125 807.671997 0.652597 1 Mancha
│ 2 1203.317749 518.720886 1300.247681 595.273621 0.552287 1 Mancha
│   xmin      ymin      xmax      ymax      confidence class name
│ 0 996.020508 559.721313 1097.136230 685.823303 0.761185 1 Mancha
│ 1 1101.128662 729.540100 1197.677612 807.828064 0.625436 1 Mancha
│ 2 1203.167603 518.520569 1299.318481 595.091675 0.568674 1 Mancha
│   xmin      ymin      xmax      ymax      confidence class name
│ 0 995.843323 559.385620 1097.216553 685.375000 0.759919 1 Mancha
│ 1 1100.075684 729.564819 1198.426147 811.619751 0.669810 1 Mancha
│ 2 1203.206177 519.212697 1299.682983 595.636108 0.548532 1 Mancha
│   xmin      ymin      xmax      ymax      confidence class name
│ 0 996.067932 559.448181 1096.566650 685.369019 0.747703 1 Mancha

```

Fig. 8 Porcentaje de seguridad

Tras hacer las pruebas correspondientes, se consiguieron los resultados de una toma de 50 muestras, las cuales se plasmaron en una matriz de confusión, es decir, una tabla en donde se puede evaluar el rendimiento del sistema de calidad desarrollado a través de la codificación en Python. Estas muestras tomadas se clasificaron por diferentes clases, las cuales detallan los tipos de aciertos o errores que puede tener el modelo realizado.

Verdadero Positivo (TP): Este viene a ser el resultado en el que el modelo detecta correctamente la mancha.

Verdadero Negativo (TN): En este tipo, se muestran los resultados donde el modelo indica de manera correcta que no existe ninguna mancha en la imagen capturada.

Falso Positivo (FP): Este hace referencia a cuando el resultado indica la presencia de una mancha en donde no hay.

Falso Negativo (FN): Este tipo de error indica el resultado en el que el modelo no detecta ningún tipo de mancha cuando en realidad sí existe.

En la Tabla II se puede analizar que de los resultados arrojados se distribuyeron de manera que hubo una cantidad de 25 muestras verdaderamente positivas, 1 falso positivo y 24 verdaderos negativos.

TABLA II  
MATRIZ DE CONFUSIÓN

Valores	Reales		
	Tipo	Positivo	Negativo
Predicción	Positivo	25	1
	Negativo	0	24

La aplicación de una matriz de confusión es importante, ya que permite evaluar y corroborar que el modelo creado se comporte de la manera deseada, Es por ello, que la precisión en este trabajo es fundamental para que se pueda tener certeza y seguridad al momento de poner en funcionamiento el sistema creado, esta métrica se fundamenta en las predicciones positivas y negativas, las cuales son derivadas de las pruebas de programación utilizando imágenes del producto, comparando los resultados obtenidos. En otras palabras, representa la relación entre las predicciones positivas del programa (correctas) y las predicciones que se realizan manualmente, permitiendo poder relacionarlas. Para ello, se muestra la fórmula (3), la cual es la correspondiente para la obtención del porcentaje de precisión.

$$Precisión = \frac{TP}{TP + FP} \dots (3)$$

Es así como en la Tabla III, se representa la aplicación de la ecuación anteriormente mencionada (3) para las diversas muestras que fueron tomadas; asimismo, como resultado se puede observar el porcentaje de precisión de cada lote de toma de muestras.

TABLA III  
TABLA DE PRECISIÓN

Muestra	Detección perfecta	Falsa detección	% Precisión
20 tomas	19	1	95%
40 tomas	39	1	97.5%
50 tomas	49	1	98%

#### IV. DISCUSIONES

El sistema diseñado para este proyecto propone una ayuda eficiente para un correcto manejo de la calidad de las conservas de alcachofa, logrando detectar las manchas generadas por roturas celulares, consiguiendo un 98% de precisión, la cual es un porcentaje bastante bueno, el cual refleja la efectividad del algoritmo desarrollado. Estos resultados muestran una mejora notable en comparación del trabajo que se hizo [11], el cual manejó el método de la regresión lineal y la agrupación K-MEANS, consiguiendo un porcentaje de precisión del 74.71%. La mejora realizada en cuestión de precisión, se debe a que no se aplicaron los métodos anteriormente mencionados, sino que se recurrió al uso de YOLOv5, el cual se basó en el método de aprendizaje automático, esta red fue alimentada de diversas imágenes donde se mostraban los diversos tipos de manchas que pueden surgir a partir de la rotura celular, logrando hacer que la red aprenda a identificar todo lo que se le ha mostrado anteriormente y que cuando se le ponga a prueba, pueda detectar exitosamente la mancha de la conserva que se está mostrando.

#### V. CONCLUSIONES

Se logró desarrollar exitosamente un sistema de control de calidad haciendo uso de las redes neuronales y la visión artificial que permitió detectar las diferentes manchas generadas por roturas celulares que se pueden encontrar en las conservas de alcachofa. Esto nos ayudó a lograr nuestro objetivo general, ya que gracias al desarrollo de este sistema las empresas evitarían pérdidas y podrían llegar a tener más ganancias.

Se investigó sobre el programa Python y la red neuronal convolucional YOLOv5, permitiendo que se pueda ver plasmado en el desarrollo de un algoritmo que fue diseñado para la detección de manchas en las conservas de alcachofa.

Se consiguió recolectar el material visual necesario, que constaba de 1054 imágenes, en las que se tenían imágenes de conservas con manchas y las que no, a través de esto se pudo crear la data de manera efectiva, la cual se usó como base para el diseño del modelo.

Asimismo, se logró realizar una codificación adecuada usando el material visual para desarrollar un buen sistema de calidad, el cual puesto en práctica nos mostró el 98% de precisión, este resultado tuvo como base las predicciones positivas y las predicciones falsas que se consiguieron a través de la ejecución del algoritmo en tiempo real.

Se concluye que el trabajo realizado tiene una mayor precisión a otras en comparación, manejando un 98%, el cual permite aumentar el nivel de confianza para su futura implementación en las fábricas procesadoras de conservas de alcachofa en el área de trabajo relacionada con la calidad, ya que minimiza el margen de error al momento del monitoreo correspondiente, además de que posibilita una reducción de costos para la empresa y así mismo beneficia a la empresa de tal manera que le permite tener más ventas.

#### REFERENCIAS

- [1] La Cámara. (2020). Importación de conservas y alimentos preparados aumentó 12%. En: La Cámara [en línea]. Publicado el 18 de mayo de 2020 [consultado el 3 de septiembre de 2023]. Disponible en: <https://lacamara.pe/importacion-de-conservas-y-alimentos-preparados-aumento-12/>
- [2] Portal Portuario. (2023). Perú: Exportaciones de alcachofas totalizan USD 162 millones en 2022. En: PortalPortuario [en línea]. Publicado el 17 de febrero de 2023 [consultado el 3 de septiembre de 2023]. Disponible en: <https://portalportuario.cl/peru-exportaciones-de-alcachofas-totalizan-usd-162-millones-en-2022/>
- [3] Evaluación de la calidad de conservas de pescado. Monografías Plus. (2020) Monografías.com. Tesis, Documentos, Publicaciones y Recursos Educativos. [en línea]. Publicado el 20 de setiembre de 2020 [consultado el 4 de septiembre de 2023]. Disponible en: <https://www.monografias.com/docs/EVALUACION-DE-LA-CALIDAD-DE-CONSERVAS-DE-FKCXQZ5YBY>
- [4] Srinath, K. (2017). Python – The Fastest Growing Programming Language. International Research Journal of Engineering and Technology. -ISSN: 2395-0056. [en línea]. Publicado en diciembre del 2017. Disponible en: <https://www.irjet.net/archives/V4/i12/IRJET-V4I1266.pdf>
- [5] León, Ryan, et al. (2022) SIMULATION OF A CANNING JAR SORTING SYSTEM BY MACHINE VISION IN ENTERPRISES USING PYTHON. [consultado el 3 de septiembre de 2023]. Disponible en: <https://laccei.org/LEIRD2022-VirtualEdition/full-papers/FP181.pdf>
- [6] Giannakourou, M & Tsironi, T. (2021). Application of Processing and Packaging Hurdles for Fresh-Cut Fruits and Vegetables Preservation. MDPI. [en línea]. Publicado el 11 de abril de 2021. Disponible en: <https://doi.org/10.3390/foods10040830>
- [7] Grattan, S & Oster, J. (2003). Water Quality Guidelines for Vegetable and Row Crops. University of California, Davis [en línea]. Publicado el 9 de septiembre de 2003 [consultado el 3 de septiembre]. Disponible en: <https://lawr.ucdavis.edu/cooperative-extension/irrigation/drought-tips/water-quality-guidelines-vegetable-and-row-crops>
- [8] CABI. (2019). Cynara cardunculus var. scolymus (alcachofa). Digital Library [en línea]. Publicado el 20 de noviembre [consultado el 3 de septiembre]. Disponible en: <https://doi.org/10.1079/cabicompendium.17585>
- [9] Fruit and Vegetable Transportation: A Comprehensive Guide. (2023). Spotos [en línea]. Publicado el 29 de julio de 2023 [consultado el 3 de septiembre de 2023]. Disponible en: <https://spotos.eu/blog/fruit-and-vegetable-transportation>
- [10] Gurovih, L. (s.f.) Fundamentos y diseño de Sistemas de Riego. Repositorio del Instituto Interamericano de Cooperación para la Agricultura [en línea]. [consultado el 3 de septiembre de 2023]. Disponible en: <https://repositorio.iica.int/bitstream/handle/11324/7213/BVE18040268e.PDF?sequence=1&isAllowed=y>
- [11] Suarez, D. (2013). Determinación de manchas en huevo de gallina mediante visión artificial. En: Repositorio UPB [en línea]. Publicado en marzo de 2013 [consultado el 3 de setiembre de 2023]. Disponible en: <http://hdl.handle.net/20.500.11912/5094>
- [12] Fernández, A. (2018). Sistema electrónico para el control de calidad de huevos de gallina mediante procesamiento de imágenes. Tesis de Licenciatura. Universidad Técnica de Ambato. Facultad de Ingeniería en Sistemas, Electrónica e Industrial. Carrera Ingeniería en Electrónica y Comunicaciones. Recuperado de <http://repositorio.uta.edu.ec/jspui/handle/123456789/28950>

- [13] Del Castillo, E. (2018). Desarrollo de un sistema de visión artificial para realizar una clasificación uniforme de limones [Tesis de licenciatura, Universidad Privada del Norte]. Repositorio de la Universidad Privada del Norte. Recuperado de: <https://hdl.handle.net/11537/13118>
- [14] La Madrid, L. (2019). Implementación de un algoritmo de control de calidad para la selección de productos agrícolas utilizando visión artificial. [Tesis de licenciatura, Universidad Nacional Pedro Ruiz Gallo]. Disponible en: <https://repositorio.unprg.edu.pe/handle/20.500.12893/4737>
- [15] Calle, L., & Torres, L. (2022). Diseño de una red neuronal convolucional para identificar la madurez adecuada del plátano para el cumplimiento de la norma técnica peruana n°011.005(2009). [Tesis para obtener título profesional, Universidad Nacional de Piura]. <https://repositorio.unp.edu.pe/bitstream/handle/20.500.12676/4305/IMEC-CAL-TOR-2022.pdf?sequence=1>
- [16] Sistema de visión artificial para la identificación del estado de madurez de frutas (granadilla). Figueroa, Diego Escobar y Guerrero, Edgar Roa. 2016, Redes de Ingeniería. <https://geox.udistrital.edu.co/index.php/REDES/article/view/10056/11679>
- [17] Marín, J. (2012). Introducción a las redes neuronales aplicadas. En: Halweb [en línea]. Publicado en febrero de 2014 [consultado el 22 de setiembre de 2023]. Disponible: <https://halweb.uc3m.es/esp/Personal/personas/jmmarin/esp/Expertos/CursoNN12.pdf>
- [18] Gonzáles, A. Ramírez, J. Vega, J. (2019). Red neuronal convolucional de bajo costo para el reconocimiento de iris. ISSN 1405-2172. Congr. Int. en Ing. Electrónica. Mem. ELECTRO. [en línea]. [consultado el 22 de setiembre de 2023]. Disponible en: [http://electro.itchiuhua.edu.mx/revista/2019/PDI-3\\_C.pdf](http://electro.itchiuhua.edu.mx/revista/2019/PDI-3_C.pdf)
- [19] Gamco. (2022). Redes Neuronales Convolucionales. En: Gamco [en línea]. Publicado el 21 de mayo de 2022 [consultado el 22 de setiembre de 2023]. Disponible en: <https://n9.cl/ejqvn>
- [20] Rozada, D. (2021). Estudio de la arquitectura YOLO para la detección de objetos mediante Deep learning. En: Repositorio de la Universidad de Valladolid [en línea]. Disponible en: <http://uvadoc.uva.es/handle/10324/45359>
- [21] Artola, A. (2019) Clasificación de imágenes usando redes neuronales convolucionales en Python. En: idus [en línea]. Publicado el 2 de junio de 2020 [consultado el 1 de octubre de 2023]. Disponible en: <https://hdl.handle.net/11441/89506>
- [22] IBM. ¿Qué son las redes neuronales convolucionales? [En línea] 2023. <https://www.ibm.com/es-es/topics/convolutional-neural-networks>.
- [23] López, F. Aviles, C. Villegas, J. (2019). Red neuronal convolucional con extracción de características multi-columna para clasificación de imágenes. Computing Science. [consultado el 22 de setiembre de 2023]. Disponible en: <http://dx.doi.org/10.13053/rcs-148-7-29>
- [24] Sarahí, S. Freire, E. (2022). Intro a las redes neuronales convolucionales. [En línea]. Disponible en: <https://bootcampai.medium.com/redes-neuronales-convolucionales-5e0ce960caf8>
- [25] Calvo, Diego. Redes neuronales. [En línea] 2017. <https://www.diegocalvo.es/red-neuronal-convolucional>.
- [26] Villada, F., Muñoz, N., & García, E. (2012). Aplicación de las Redes Neuronales al Pronóstico de Precios en el Mercado de Valores. CIT Información Tecnológica, 23(4), 11–20. <https://doi.org/10.4067/s0718-07642012000400003>
- [27] Lozada, W. Suarez, M. Avendaño, E. (2021). Aplicación de redes neuronales convolucionales para la detección del tizón tardío Phytophthora infestans en papa Solanum tuberosum. Revista UDCA Actualidad & Divulgación Científica [en línea]. Publicado el 13 de octubre de 2021 [consultado el 3 de setiembre de 2023]. Disponible en: <http://dx.doi.org/10.31910/rudca.v24.n2.2021.1917>