

REVISTA CIENTÍFICA  
**CONECTIVIDAD**

Conectividad  
ISSN: 2806-5875  
ISSN-L: 2806-5875  
revista@ister.edu.ec  
Tecnológico Superior Rumiñahui  
Ecuador

Villarreal Ger, Lucio Orlando  
Prototipo para la detección y clasificación de productos  
alimenticios mediante visión artificial en base al color  
Conectividad, vol. 5, núm. 2, Esp., 2024, pp. 46-62  
Tecnológico Superior Rumiñahui  
Sangolquí, Ecuador

DOI: <https://doi.org/10.37431/conectividad.v5i2.129>

Disponible en: <https://www.redalyc.org/articulo.oa?id=777882631004>

- ▶ [Cómo citar el artículo](#)
- ▶ [Número completo](#)
- ▶ [Más información del artículo](#)
- ▶ [Página de la revista en redalyc.org](#)

redalyc.org

Sistema de Información Científica Redalyc  
Red de revistas científicas de Acceso Abierto diamante  
Infraestructura abierta no comercial propiedad de la academia

## **Prototipo para la detección y clasificación de productos alimenticios mediante visión artificial en base al color**

*Prototype for the detection and classification of food products using artificial vision based on color*

Lucio Orlando Villarreal Ger<sup>1</sup> 

<sup>1</sup>Instituto Superior Tecnológico Vicente Fierro, lvillarreal@institutovicentefierro.edu.ec,  
Tulcán, Ecuador

Autor para correspondencia: lovgycea@gmail.com

**Fecha de recepción:** 2023.11.24

**Fecha de aceptación:** 2024.01.19

**Fecha de publicación:** 2024.02.20

### **RESUMEN**

El presente trabajo muestra un prototipo que aplica la visión artificial como una herramienta útil en los procesos de manufactura que requieren clasificar sus productos, ya sea por defectos, cumplimiento de normas, por característica como peso, contorno, color, en general para el control de la calidad de los productos. El proyecto tiene como objetivo desarrollar un prototipo para la detección y clasificación de productos alimenticios mediante visión artificial tomando como referencia el color. Para el desarrollo del proyecto, al ser de tipo experimental, se procede a escoger los componentes para el diseño del prototipo, así como la característica del objeto que se requiere controlar, en este caso, el color; la siguiente etapa es configurar un algoritmo que permita que el prototipo detecte la característica elegida del objeto; para este propósito, el lenguaje de programación utilizado es Python, que se rige bajo licencia de software libre y cuenta entre otras librería, OpenCV, una biblioteca de visión por computador de código abierto que tiene 500 funciones y alrededor de 2500 algoritmos; en la siguiente fase se integran los componentes del prototipo: cámara web, fuente de iluminación, computador o mini controlador, software de procesamiento de imágenes, pantalla de visualización. Se realiza pruebas en tiempo real para validar el prototipo, resultando en un prototipo funcional de acuerdo al objetivo planteado. Esto permite concluir que los sistemas de detección de defectos o de control de calidad basados en visión artificial pueden adaptarse a los procesos de manufactura, mejorando su productividad y competitividad.

**Palabras clave:** Algoritmo, Control de calidad, Prototipo, Visión artificial.

### **ABSTRACT**

This work showcases a prototype that uses artificial vision to enhance manufacturing processes that require the classification of products. This can include detecting defects, ensuring compliance with standards, and categorizing products based on characteristics such as weight, contour, and color. The project aims to develop a prototype for the detection and classification

of food products using artificial vision, where color serves as a reference point. To develop the prototype, the components for the design are chosen, and the characteristics of the object to be controlled are identified. In this case, the chosen characteristic is color. An algorithm is then created to enable the prototype to detect the object's feature. For this purpose, the programming language Python is used, which is governed by a free software license and has a library called OpenCV. OpenCV is an open-source computer vision library that contains around 2,500 algorithms and 500 functions. In the next stage, the prototype components, including a web camera, lighting source, computer or mini controller, image processing software, and display screen, are integrated. Real-time tests are conducted to validate the prototype, resulting in a functional prototype that meets the stated objective. This proves that artificial vision-based quality control systems can be adapted to manufacturing processes, leading to improved productivity and competitiveness.

Keywords: Algorithm; Quality control; Prototype; Computer vision.

## **INTRODUCCIÓN**

En la actualidad las industrias que fabrican productos buscan estar a la vanguardia en el uso de tecnologías que mejoren la calidad de sus procesos y de sus productos, reducir costos y aumentar las utilidades, una de las soluciones tecnológicas que más evolución ha tenido en los últimos años son los sistemas de visión artificial por computadora (Icaza, 2019); en el caso de procesos industriales por ejemplo, éstos generalmente requieren la automatización de un sistema de visión artificial, para verificar que los productos manufacturados, frutas, vegetales cumplan con ciertos criterios de calidad previamente establecidos, evitando los defectos de producción, impurezas no deseadas, mala apariencia de los productos (Barriga, 2006; De la Fuente, 2012).

La inspección visual manual es un método comúnmente utilizado para detectar defectos en los productos, pero puede ser costoso y lento. Además, puede haber errores humanos que afecten la precisión de la inspección (Alpízar & Fernández, 2021). Por lo tanto, el control de calidad en los procesos de producción, es un requisito indispensable sino básico en la actualidad, Como parte de la inteligencia artificial, la visión artificial permite mejorar el control en la calidad de los productos de una manera rápida y precisa, además de reducir costos, en comparación con la inspección visual.

Mediante el presente trabajo se pretende desarrollar un prototipo para la detección y clasificación de productos alimenticios mediante visión artificial tomando como referencia el color. Se eligió

como característica a detectar el color con fines didácticos, ya que cualquier característica dependiendo del tipo de negocio puede ser configurada para ser detectada. Lo importante es la selección de los componentes adecuados tanto hardware como software, de manera que puedan integrarse en un solo sistema que funcione de manera coordinada con el sistema de producción y cuyos resultados puedan obtenerse en tiempo real.

El potencial que tiene la visión artificial para su uso en diferentes ámbitos de la industria y el comercio, está fundamentado en la experiencia que tienen las grandes industrias y cuyos beneficios han sido evidentes en varios aspectos, entre los más relevantes podemos citar:

Mayor precisión y velocidad que la inspección visual manual; reducción de costos: pues la detección automática de defectos ayuda a prevenir la producción de productos defectuosos, lo que reduce los costos de retrabajo y los costos de garantía (Alpízar y Fernández, 2021); mejora de la eficiencia: la visión artificial permite la inspección en tiempo real y la detección rápida de defectos, lo que mejora la eficiencia del proceso de control de calidad y ayuda a reducir el tiempo de inactividad (Alpízar y Fernández, 2021); flexibilidad: la visión artificial puede ser configurada para detectar una amplia gama de defectos y puede ser adaptada a diferentes tipos de productos y procesos de manufactura (Ayo y Moreno, 2021); mejora de la calidad del producto: la implementación del control de calidad mediante visión artificial ayuda a garantizar que los productos fabricados cumplan con las especificaciones de diseño y los requisitos de calidad, lo que mejora la calidad del producto y la satisfacción del cliente (Olano, 2019).

Los conocimientos que se requiere para el diseño de un sistema de control de calidad basado en visión artificial, caen en el ámbito de la informática y electrónica, disciplinas que son parte de la malla curricular de la carrera de Tecnología Superior en Electricidad del Instituto Superior Tecnológico Vicente Fierro.

Para el desarrollo de este proyecto se utilizó el lenguaje de programación Python 3.10.10 el cual incorpora librerías diseñadas para este tipo de proyectos. Según Del Pianta (2020), Python es uno de los lenguajes de programación más populares y versátiles en la actualidad, especialmente para proyectos de inteligencia artificial y visión por computadora, cuenta con una gran cantidad de bibliotecas de visión por computadora como OpenCV, Scikit-image, Pillow, entre otras, que facilitan el procesamiento de imágenes y el reconocimiento de patrones. Estas bibliotecas permiten la detección de colores, formas y objetos en las imágenes, lo que resulta útil en un

sistema de control de calidad basado en visión artificial.

Asimismo, Python se puede integrar fácilmente con otros lenguajes de programación y herramientas, lo que lo hace ideal para proyectos que requieren la colaboración de diferentes equipos y tecnologías (Ayo y Moreno, 2021), en el presente trabajo, la integración con el microprocesador Arduino, no requirió de mayor complejidad.

El prototipo diseñado, permite tener una clara comprensión del potencial de uso de la visión artificial, en las pequeñas y medianas empresas que existen en la zona de influencia del IST Vicente Fierro, a costos relativamente asequibles, esto debido a que en la actualidad existe una gran variedad de aplicaciones de licencia libre que permiten integrarse con cualquier sistema de producción o de control de procesos, de acuerdo a las necesidades del negocio.

Al diseñar un sistema de visión artificial, resulta imprescindible la realización de pruebas a fin de calibrar los instrumentos y componentes, bajar la incertidumbre en la detección de defectos, para garantizar el control de calidad de los productos.

El objetivo general del presente trabajo fue desarrollar un prototipo para la detección y clasificación de productos alimenticios mediante visión artificial tomando como referencia el color. Entre los objetivos específicos que se establecieron están: configurar el algoritmo de detección de característica del producto, integrar los componentes del sistema de visión artificial para la detección de características del producto y verificar la eficacia del prototipo de visión artificial mediante la realización de pruebas.

## **METODOLOGÍA Y MATERIALES**

Para llevar a cabo este proyecto de tipo experimental, las siguientes fases fueron necesarias:

Fase 1: elección de los componentes del sistema y del software

Según Mendoza y Salazar (2019), Para diseñar un sistema de control de calidad en un proceso de manufactura mediante visión artificial básico, se pueden utilizar los siguientes componentes: cámara, sistema de iluminación, computadora o microcontrolador, software de procesamiento de imágenes, pantalla de visualización. Se describe a continuación las variables que deben considerarse para la elección de cada uno de los componentes del sistema:

Cámara: para capturar imágenes de los productos durante el proceso de fabricación, algunas de

las características que deben considerarse al elegir una cámara para este propósito son: resolución: la resolución de la cámara debe ser suficientemente, alta para capturar imágenes de alta calidad y permitir la detección de defectos. Velocidad de captura: la cámara debe ser capaz de capturar imágenes con la suficiente frecuencia para permitir el control en tiempo real del proceso de manufactura. Sensibilidad a la luz: la sensibilidad a la luz de la cámara debe ser adecuada para la iluminación disponible en el entorno de trabajo. Tamaño y formato de la cámara: el tamaño y formato de la cámara deben ser compatibles con las necesidades específicas de la aplicación o el software utilizado. Algunos tipos comunes de cámaras utilizadas para el control de calidad en proceso de manufactura mediante visión artificial incluyen cámaras CCD (dispositivo de carga acoplada) y cámaras CMOS (sensor de imagen complementario de metal-óxido-semiconductor). Las cámaras de alta velocidad y las cámaras infrarrojas también pueden ser utilizadas en algunas aplicaciones específicas (Del Pianta, 2020).

Iluminación: para mejorar la calidad y claridad de las imágenes capturadas por la cámara; una iluminación adecuada puede mejorar la capacidad de la cámara para detectar y clasificar los objetos según su calidad, ya que puede resaltar las características importantes de los objetos y permitir una mejor diferenciación entre los objetos buenos y defectuosos (Del Pianta, 2020). Existen diferentes tipos de iluminación, como la iluminación de luz difusa, la iluminación de luz directa, la iluminación de luz polarizada y la iluminación de luz infrarroja. Cada tipo de iluminación tiene sus propias ventajas y desventajas, y debe ser seleccionado de acuerdo con las necesidades específicas de la aplicación y las características de los objetos a inspeccionar (Del Pianta, 2020). En general, es proporcionar una iluminación uniforme y controlada, que permita obtener imágenes de alta calidad y precisión para la inspección y clasificación de objetos en la línea de producción. Computadora o microcontrolador: para procesar las imágenes capturadas y llevar a cabo los algoritmos de visión artificial necesarios para detectar defectos y clasificar los productos., la elección del tipo de computadora o microcontrolador depende del alcance y complejidad de la aplicación de control de calidad: para aplicaciones de visión artificial más sencillas, como el control de calidad de piezas pequeñas y la clasificación de productos según su forma o color, se pueden utilizar microcontroladores de bajo costo como el Arduino o el Raspberry Pi, que son fáciles de programar y pueden realizar tareas de procesamiento de imágenes en tiempo real (Del Pianta, 2020). Para aplicaciones más complejas que involucran un mayor procesamiento de imágenes y análisis de datos, se pueden utilizar computadoras con una

mayor capacidad de procesamiento y memoria, como las PC industriales o las computadoras embebidas. Estas computadoras son capaces de manejar grandes cantidades de datos en tiempo real y realizar análisis de imágenes más complejos. Es importante seleccionar el tipo de hardware adecuado para garantizar un rendimiento óptimo y una precisión en la inspección y clasificación de los objetos (Mendoza y Salazar, 2019).

**Software:** Para procesar las imágenes y llevar a cabo las operaciones de segmentación, detección de bordes y clasificación, el software que permite enlazar y ejecutar la programación del sistema, debe considerar lo siguiente: procesamiento de imágenes: se utiliza para capturar imágenes de los productos en el proceso de fabricación y procesarlas para detectar defectos o variaciones. Análisis de imagen: se utiliza para analizar y medir características específicas de los productos, como dimensiones, forma, color, textura y patrones. Reconocimiento de patrones: se utiliza para detectar patrones y anomalías en las imágenes de los productos y en los datos de producción. Aprendizaje automático: se utiliza para entrenar a los sistemas de visión artificial en la detección de defectos y en la toma de decisiones basadas en los datos de producción (Del Pianta, 2020). En general, la combinación de estos diferentes tipos de software permite el control de calidad en el proceso de manufactura, de manera más eficiente y precisa y por lo tanto la satisfacción del cliente. Hay algunos softwares de visión artificial gratuitos que se pueden utilizar, aunque su funcionalidad y capacidades pueden ser más limitadas que las de las soluciones comerciales. Algunos ejemplos son: OpenCV: es una biblioteca de código abierto que se utiliza para el procesamiento de imágenes y la visión artificial en una amplia variedad de aplicaciones (De la Torre, 2022). Si bien no está diseñada específicamente para el control de calidad en la manufactura, se puede utilizar para desarrollar algoritmos de detección de defectos y análisis de imágenes (Ayo y Moreno, 2021). ImageJ: es un programa gratuito y de código abierto para el procesamiento y análisis de imágenes, que se utiliza en una amplia variedad de aplicaciones de investigación científica y médica. También se puede utilizar para el análisis de imágenes en el control de calidad en la manufactura (Del Pianta, 2020). VVVV: es una herramienta de programación visual gratuita y de código abierto que se utiliza para el desarrollo de aplicaciones de visión artificial y multimedia. Puede ser utilizado para desarrollar sistemas de control de calidad en la manufactura mediante la integración de cámaras y algoritmos de procesamiento de imágenes (Del Pianta, 2020). Processing: es un software de programación gráfica utilizado para el desarrollo de aplicaciones creativas, educativas y científicas. Processing se puede utilizar con

Arduino para desarrollar soluciones de visión artificial para el control de calidad (Ayo y Moreno, 2021).

Pantalla de visualización: para mostrar los resultados del proceso de inspección y control de calidad. La pantalla de visualización en el proceso de inspección y control de calidad es un componente importante que se utiliza para mostrar los resultados del proceso de inspección y control de calidad. Es una pantalla de visualización en tiempo real que muestra las imágenes capturadas por el sistema de visión artificial y los resultados de análisis de imágenes, permitiendo a los operadores y técnicos ver los resultados del proceso de inspección de manera clara y fácil de interpretar (Del Pianta, 2020), la pantalla de visualización puede mostrar una variedad de información, incluyendo imágenes en tiempo real de los productos en proceso de fabricación, los resultados de análisis de imágenes, los datos de calidad y cualquier otro tipo de información relevante para el control de calidad en el proceso de manufactura (Mendoza y Salazar, 2019). La pantalla de visualización puede ser una parte importante de un sistema de visión artificial más amplio, que puede incluir hardware de captura de imágenes, software de procesamiento de imágenes y algoritmos de análisis de imágenes. En general, una pantalla de visualización efectiva debe ser fácil de usar, precisa y clara, permitiendo a los operadores y técnicos tomar decisiones informadas sobre el proceso de fabricación y el control de calidad (Ayo y Moreno, 2021).

## **Fase 2: Desarrollo del código de programación mediante PYTHON**

Código de programación en Python

```
import cv2
import serial
# Configurar la conexión con el Arduino
arduino = serial.Serial('COM4', 9600)
# Configurar la cámara web
cap = cv2.VideoCapture(1)
# Definir los rangos de color que se van a detectar (en este caso, verde)
lower_green = (36, 25, 25)
upper_green = (70, 255, 255)
while True:
    # Capturar un fotograma de la cámara
    ret, frame = cap.read()
    # Redimensionar el fotograma para que sea más fácil de procesar
    frame = cv2.resize(frame, (640, 480))
    # Convertir el fotograma a espacio de color HSV
    hsv = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)
    # Aplicar un filtro para detectar los colores en el rango especificado
    mask = cv2.inRange(hsv, lower_green, upper_green)
```

```
# Calcular el área del objeto detectado
contours, hierarchy = cv2.findContours(mask, cv2.RETR_TREE,
cv2.CHAIN_APPROX_SIMPLE)
# Si se detecta un objeto con un área mayor que 1000 píxeles, encender la luz verde del
Arduino
# Si no se detecta ningún objeto o se detecta un objeto que no es de color verde, encender la
luz roja del Arduino
is_green = False
for cnt in contours:
    area = cv2.contourArea(cnt)
    if area > 1000:
        is_green = True
    if is_green:
        arduino.write(b'1')
else:
    arduino.write(b'2')
# Mostrar el fotograma con el resultado del filtro
cv2.imshow('frame_proyecto calidad ISTV', mask)
cv2.imshow("camara usb_proyecto calidad ISTV", frame)
# Salir del bucle si se presiona la tecla 'q'
if cv2.waitKey(1) & 0xFF == ord('q'):
    break
# Liberar la cámara y cerrar la ventana
cap.release()
cv2.destroyAllWindows()
```

### Fase 3: Desarrollo del código de programación en Arduino e integración con Python

Código de programación en ARDUINO

```
// Definir los pines de la luz azul y la luz roja
int blueLed = 9;
int redLed = 10;
void setup() {
    // Configurar los pines como salidas
    pinMode(blueLed, OUTPUT);
    pinMode(redLed, OUTPUT);
    // Configurar la comunicación serial a 9600 baudios
    Serial.begin(9600);
}
void loop() {
    // Leer los comandos desde la comunicación serial
    if (Serial.available() > 0) {
        int command = Serial.read();
        // Encender la luz azul si el comando es '1'
        if (command == '1') {
            digitalWrite(blueLed, HIGH);
            digitalWrite(redLed, LOW);
        }
    }
}
```

```
// Encender la luz roja si el comando es '2'  
else if (command == '2') {  
    digitalWrite(blueLed, LOW);  
    digitalWrite(redLed, HIGH);  
}  
}
```

#### **Fase 4: Desarrollo de un prototipo basado en visión artificial y microcontroladores.**

Una vez que se completa la programación del sistema, se procedió a desarrollar el prototipo de visión artificial utilizando los componentes descritos, una cámara USB 2mpx, un microprocesador Arduino, una protoboard, un computador y para las pruebas: manzanas de diferentes colores. El objetivo era que el software de visión artificial capturara la imagen, la analizara y enviara una señal al microprocesador para ejecutar un proceso, en este caso, encender dos diodos emisores de luz de color rojo y verde.

Cuando se detecta una manzana de color verde, el sistema de visión artificial, basado en programación Python y la librería Open\_cv, envía una señal al microprocesador a través del puerto serial con una velocidad de transmisión de 9600 baudios para encender el diodo verde. En caso contrario, si no hay objeto o la manzana es de otro color, se enciende el diodo de color rojo. Además, se proporciona una señal de visión en la CPU para el operador, tanto en fotograma como en la imagen real.

Se procedió a implementar un circuito básico de control mediante, el microcontrolador Arduino, el mismo que ejecuta el proceso de control, en este caso encendido de dos salidas. El proceso es totalmente automático y se puede salir del aplicativo utilizando la letra "q" del teclado. Este prototipo puede ser útil para la evaluación de una amplia variedad de alimentos, desde frutas y verduras hasta carnes y productos lácteos. Además, su diseño modular permite su adaptación a diferentes necesidades y requisitos específicos de cada tipo de alimento. En resumen, un prototipo de control de calidad basado en visión artificial y microcontroladores puede ser una herramienta valiosa para garantizar la calidad y la seguridad de los alimentos. Su capacidad para detectar automáticamente el color del alimento y tomar decisiones en tiempo real puede ahorrar tiempo y reducir el riesgo de error humano en el proceso de control de calidad.

#### **Materiales**

Los materiales usados se presentan en la siguiente tabla:

**Tabla 1.** Componentes del sistema.

<b>Nombre del componente</b>	<b>Código</b>
Microprocesador	ATmega 328P
CPU CORE I5 ENVIDIA 8MB 11GN	CPU -I5
Cámara CMOS 1080p	CAM WEB -USB
LUCES LED	LED 5V RGB
FUENTE DE VOLTAJE 2AMP	F-2AMP-ENFORC
Módulo relé de 5 voltios	SRD-05VDC-SL-C

**Tabla 2.** Materiales utilizados en el proyecto.

<b>Cantidad</b>	<b>Material (dispositivo)</b>
1	Tarjeta arduino uno
1	Módulo relé SRD-05VDC-SL-C
1	Cámara WEB-USB 2MPX
1	Tcrt5000
1	Fuente de voltaje 2 AMP
1	CPU core I5 para ejecución de software
1	Cable UTP
1	Cinta aislante

## RESULTADOS Y DISCUSIÓN

Al ser un trabajo de tipo experimental, los resultados obtenidos podemos dividirlos en tres partes, esto debido a que en una primera fase se procede a la instalación del programa Python y al diseño del algoritmo correspondiente en este programa, se realizan las corridas correspondientes, hasta tener el código de programación de acuerdo a lo requerido. La ejecución de este código puede observarse en la figura No.1. En una segunda fase se desarrolla el código de programación en Arduino y su integración con Python, una vez realizadas las corridas correspondientes, y los ajustes necesarios, se tiene la ejecución de este código como puede observarse en la figura No.2. En una tercera fase se integran los diferentes componentes tanto hardware como software y se realizan las pruebas correspondientes cuyos resultados se detallan en la tabla No.1



No.	Descripción	Resultado	Observación
1	Funcionalidad programa Python	Se ejecuta a satisfacción	Python 3.10.10
2	Instalación librerías Pyserial y Open CV	Instaladas	Las librerías permiten realizar la integración del sistema con los periféricos.
3	Establecer comunicación entre Python y Arduino	Se realiza el RUN en Python y se establece comunicación a 9600 baudios.	Puerto configurado COM4
4	Establecer comunicación con la cámara WEB	El sistema se comunica con la cámara WEB	La cámara WEB utilizada para la visión artificial
5	Ejecución del sistema total	El programa reconoce los alimentos de color verde en este caso manzanas	Se configuró para que el sistema reconozca alimentos de color verde
6	Ejecución del sistema total.	El programa reconoce otros alimentos de otros colores	Cuando los alimentos detectados son de otro color, se enciende una luz roja.

Se escogió como característica a detectar del producto (manzana), el color, si es verde el sistema enciende una luz verde, caso contrario una luz roja; al tener un resultado positivo con esta característica, se puede inferir que se puede configurar el sistema para cualquier otra característica o defecto que se desee controlar. Los componentes del sistema de visión artificial, deben tener ciertas características que permitan integrarse entre sí y con otros sistemas, en el presente trabajo, se eligió el programa Python y Open CV que son software de licencia libre y como microcontrolador el Arduino, que ofrecen gran versatilidad a la hora de configurar un sistema.

Los resultados obtenidos con el prototipo, permiten visualizar un gran campo de aplicación de la visión artificial en las pequeñas y medianas empresas, que requieran optimizar su producción o mejorar la calidad de sus productos, tal como se demuestra en el trabajo de tesis realizado por Ramos J, sistema de visión artificial para el conteo y medición de alevinos de trucha “arco iris” para la dirección subregional de la producción Andahuaylas.

Es necesario mencionar que, para tener el prototipo operativo, fueron necesarias varias pruebas tanto a nivel de programación como a nivel de hardware, hasta que la configuración final produzca los resultados deseados, estos es la detección de la característica del objeto de estudio, esto es concordante con los resultados obtenidos por Icaza W, en su trabajo Desarrollo de un

sistema de visión artificial con LabVIEW, que permita detectar imperfecciones en las latas de atún.

Se conoce que a nivel industrial el control de calidad de los productos es un proceso clave, que permite la detección temprana de defectos en los productos, los resultados del presente trabajo demuestran que es posible la aplicación de la visión artificial a pequeña y mediana escala y a costos relativamente bajos en comparación con los de las grandes industrias. Los costos reducidos se deben en gran parte a que existen software de licencia libre que pueden utilizarse en las pequeñas industrias o negocios, esto puede corroborarse con los resultados obtenidos por Rosas-Echevarría, C.W.; Solís-Bonifacio, H.; Cerna-Cueva, A.F. (2019). Sistema eficiente y de bajo costo para la selección de granos de café: una aplicación de la visión artificial: “con respecto a los costos, también resultan muy bajos en comparación con los ojos electrónicos, ya que los equipos son de bajo costo y el software es de código libre (OpenCV). El presente sistema es flexible para control de calidad con parámetros estandarizados, ya que el sistema recibe los valores estándar que deseemos para la posterior selección”.

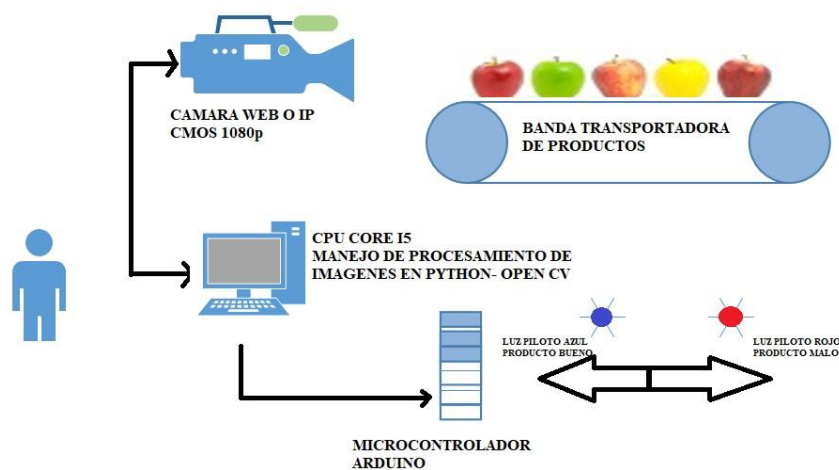


Figura 3. Sistema planteado como solución

## CONCLUSIONES

El prototipo desarrollado cumple con el objetivo planteado, esto es la detección de una característica de un producto, en este caso el color, demostrando así que los sistemas de control de calidad que involucran visión artificial, para la detección de defectos o alguna característica de un producto, son precisos y eficaces.

Se configura el algoritmo de detección del color en el programa Python, y librería Opencv de acuerdo a lo planificado. La selección y configuración del algoritmo en función de las características del producto es la clave en el desarrollo de un sistema automatizado de visión artificial, para garantizar la detección de defectos o característica del producto.

La integración de los componentes una vez desarrollados los algoritmos en Python y en el Arduino, cumple con el objetivo planteado, demostrando que el sistema de visión artificial funciona, este puede integrarse al proceso de fabricación, para la detección automática de defectos o características de los productos, esta integración puede hacerse también a otros sistemas de control de procesos para garantizar la calidad de los productos.

La optimización del sistema de visión artificial es crucial para una detección precisa de defectos, esto se lo realiza ajustando ya sea la configuración del sistema, así como la elección de los componentes más adecuados, en cuanto a iluminación o cámaras, por ejemplo.

La realización de pruebas con un prototipo de un sistema de visión artificial es necesario antes de implementar en un proceso de fabricación y/o producción, a fin de ajustar las variables requeridas dentro del rango deseado, asimismo para determinar costos.



**Figura 4.** Prototipo completo funcionando, control de calidad basado en el parámetro color (verde).

Prototipo funcionando de acuerdo a los objetivos planteados. Al pasar la manzana verde por la cámara web, una luz indicadora de color verde se enciende.



**Figura 5.** Prototipo completo funcionando: con producto de diferente color.

Prototipo con todos los elementos funcionando, en el presente estudio la acción del controlador es la emisión de una señal luminosa, pero puede configurarse para que se realice otras acciones como, por ejemplo: activación de brazos robóticos, apertura de puertas para productos de desecho, activación de alarmas sonoras, etc.

## REFERENCIAS

- Alpízar, J., & Fernández, M. (2021). Caracterización morfológica de un lecho de recubrimiento de tabletas farmacéuticas mediante un algoritmo de visión artificial. *Tecnología en Marcha*, 34(3), 51-60. doi: <https://doi.org/10.18845/tm.v34i3.5032>
- Ayo, W., & Moreno, H. (2021). Desarrollo de un sistema de inspección automático de PCB'S mediante visión artificial. [tesis de pregrado, Universidad Politécnica Salesiana]. Obtenido de <https://dspace.ups.edu.ec/bitstream/123456789/20029/1/UPS%20-%20TTS345.pdf>
- Del Pianta, L. (2020). Diseño e implementación de un sistema de control de calidad mediante visión por computador. [tesis de pregrado, Universidad Politécnica de Valencia]. Obtenido de <https://riunet.upv.es/bitstream/handle/10251/149784/Pianta%20-%20Dise%C3%B1o%20e%20implementaci%C3%B3n%20de%20un%20sistema%20de%20control%20de%20calidad%20mediante%20vision%20por%20computador.pdf?sequence=1>
- López, F. (2019). Clasificación de imágenes usando redes neuronales convolucionales. [tesis de maestría, Universidad Autónoma Metropolitana Azcapotzalco]. Obtenido de <http://hdl.handle.net/11191/6123>

- Mendoza, D., & Salazar, J. (2019). Sistema de clasificación y control de calidad en un proceso de producción industrial usando visión artificial. [tesis de maestría, Universidad Técnica de Ambato]. Obtenido de <https://repositorio.uta.edu.ec/jspui/handle/123456789/29175>
- Merino, Y. (2021). Visión artificial para el control de calidad de un retrovisor. [tesis de pregrado, Universidad Central de Catalunya]. Obtenido de <http://hdl.handle.net/10854/7003>
- Olano, W. (2019). Clasificación no destructiva de frutas utilizando inteligencia artificial: Revisión Bibliográfica. [tesis de pregrado, Universidad señor de Sipán]. Obtenido de <https://repositorio.uss.edu.pe/bitstream/handle/20.500.12802/6215/Olano%20Chavez%20Wilfredo%20Cristobal.pdf?sequence=1&isAllowed=y>
- Sedeño, J. (2021). Implementación de un prototipo con sistema de visión artificial para el procesamiento de imágenes que permita mejorar la productividad en el proceso de corte por plasma. [Tesis de pregrado, Escuela Superior Politécnica de Chimborazo]. Obtenido de <http://dspace.esepoch.edu.ec/handle/123456789/15856>.
- De La Torre, S. (2022). Sistema de clasificación de huevos mediante un algoritmo de visión artificial. [Tesis de pregrado, Universidad Técnica del Norte]. Obtenido del [repositorio.utn.edu.ec](http://repositorio.utn.edu.ec)
- López, R. & Icaza, W. (2019). Desarrollo de un sistema de visión artificial con Labview, que permita detectar imperfecciones en las latas de atún. [Tesis de pregrado, Universidad Estatal de Milagro]. Obtenido de: <http://repositorio.unemi.edu.ec/xmlui/handle/123456789/4921>
- Ana Gonzáles Marcos...[et al.] (2006). Técnicas y algoritmos básicos de visión artificial. [Material didáctico. Ingenierías. Universidad de la Rioja]. Obtenido de: <https://investigacion.unirioja.es/documentos/5c13b22ac8914b6ed3778a6a>
- Ramos, J. (2018). Sistema de visión artificial para el conteo y medición de alevinos de trucha “arco iris” para la dirección subregional de la producción Andahuaylas. [Tesis de pregrado, Universidad Nacional José María Arguedas]. Obtenido de: <https://hdl.handle.net/20.500.14168/431>
- Rosas-Echevarría, C.W.; Solís-Bonifacio, H.; Cerna-Cueva, A.F. (2019). Sistema eficiente y de bajo costo para la selección de granos de café: una aplicación de la visión artificial. *Scientia Agropecuaria* 10(3): 347 – 351. Obtenido de: <http://dx.doi.org/10.17268/sci.agropecu.2019.03.04>

....