

UNIVERSIDAD SANTO TOMÁS

Seccional Tunja

FACULTAD DE INGENIERÍA ELECTRÓNICA

TRABAJO DE INVESTIGACIÓN:

**Algoritmo de Detección, Seguimiento y
Conteo de Fresas en Secuencias de
Video con Entorno Controlado para
Cálculo de Madurez**

Desarrollado por

Javier Arévalo López

Dirigido por

Ing. Camilo Ernesto Pardo Beainy Esp. Msc. PhD(c)

Ing. Edgar Andrés Gutiérrez Cáceres Esp. Msc. PhD(c)

Mayo 2024

Algoritmo de Detección, Seguimiento y Conteo de Fresas en Secuencias de Video con Entorno Controlado para Cálculo de Madurez

Desarrollado por

Javier Arévalo López

Dirigido por

Ing. Camilo Ernesto Pardo Beainy Esp. Msc. PhD(c)

Ing. Edgar Andrés Gutiérrez Cáceres Esp. Msc. PhD(c)

Notas de Aceptación

	Firma del director
	Firma del codirector
	Firma del jurado
	Firma del jurado

Exoneración de Responsabilidades

Los conceptos, investigaciones, prácticas, análisis y conclusiones desarrolladas en este proyecto son de exclusiva responsabilidad material del autor. Esto no representa la opinión de la Universidad Santo Tomás ni la de la Facultad de Ingeniería Electrónica.

Andrés Javier Arévalo López

Resumen

Este trabajo presenta una propuesta para el monitoreo de cultivos de fresas mediante una herramienta que mejora la eficacia y eficiencia en la gestión de los cultivos. A través del uso de grabaciones de video de fresas en camas de cultivo o cintas transportadoras, la herramienta propuesta busca reducir la carga administrativa de seguir manualmente los períodos de maduración y el conteo de fresas. La herramienta emplea algoritmos para generar archivos digitales que proporcionan a los horticultores datos oportunos y accesibles sobre sus cultivos. Las características clave de la herramienta incluyen la identificación de frutas dentro de los fotogramas de video, el seguimiento consistente de cada objeto identificado a lo largo de la secuencia para mantener la precisión, y la extracción de estos objetos para el conteo y la evaluación de la madurez entre otras métricas. Esta innovación podría aumentar significativamente la producción y establecer estándares de calidad más altos.

Índice general

1. Problemática	7
1.1. Formulación de preguntas	7
1.2. Definición del problema	7
1.3. Delimitación del problema	7
2. Justificación	9
3. Objetivos	10
3.1. Objetivo General	10
3.2. Objetivos Específicos	10
4. Marco Teórico	11
4.1. Reconocimiento de Imágenes y Visión Artificial	11
4.1.1. Espacios de Color y el Modelo HSV	12
4.1.2. Enmascaramiento	14
4.2. Seguimiento de Objetos Fotograma por Fotograma	14
4.2.1. Algoritmos de Distancia Euclidiana	14
4.2.2. Intersección Sobre Unión	15
4.3. Maduración de las Fresas y Segmentación Cromática	16
4.3.1. Clusterización	16
4.4. Estado del Arte	16
5. Método Experimental y de Investigación	19
5.1. Diseño Metodológico	19
5.1.1. Tipo de Investigación	19
5.1.2. Recolección y Procesamiento de Datos	19
5.1.3. Análisis de Datos	20
5.2. Etapas y Actividades	20
5.2.1. Etapa de Preparación y Presentación de la Propuesta.	20
5.2.2. Etapa de Investigación Preliminar.	21
5.2.3. Etapa de Diseño del Algoritmo.	21
5.2.4. Etapa de Implementación, Pruebas y Ajustes.	21
5.2.5. Etapa de Documentación, Preparación de Informes y Divulgación.	21
5.3. Recursos y Métodos	22
5.3.1. Materiales y Presupuesto	22
5.3.2. Institucionales	22
5.4. Base Teórica y Simulaciones	22
5.4.1. Algoritmo de Obtención de Máscaras	23

5.4.2.	Algoritmo de Identificación, Seguimiento y Extracción de Objetos	23
5.4.3.	Algoritmo de Cálculo del estado de Madurez	25
5.4.4.	Algoritmos de Seguimiento	26
5.4.5.	Algoritmo de Limpieza de Conjuntos de Datos	27
5.5.	Detalles Experimentales	27
5.5.1.	Algoritmo de Obtención de Máscaras	28
5.5.2.	Algoritmo de Identificación, Seguimiento y Extracción de Objetos	29
5.5.3.	Algoritmo de Cálculo del estado de Madurez	30
6.	Resultados	32
6.1.	Resultados del Algoritmo de Obtención de Máscaras	32
6.2.	Resultados del Algoritmo de Identificación, Seguimiento y Extracción de Objetos	35
6.2.1.	Centroid Tracker	36
6.2.2.	Bounding Box Tracker	38
6.2.3.	Intersection Over Union Tracker	38
6.3.	Resultados Algoritmo de Cálculo del estado de Madurez	38
7.	Discusión	41
7.1.	Clasificación de las Técnicas de Detección y Seguimiento	41
7.2.	Acerca del Cálculo de Madurez	41
7.3.	Límites Experimentales	42
8.	Conclusiones	43
8.1.	Eficiencia de los Algoritmos de Seguimiento	43
8.2.	Viabilidad del Cálculo de Madurez Basado en Color	43
8.3.	Retos y Limitaciones Experimentales	43

Abstract

This work presents a proposal for monitoring strawberry crops through a tool that enhances the efficacy and efficiency of crop management. By the use of video footages of strawberries on cultivation beds or conveyors belts, the proposed tool aims to reduce the administrative burden of manually tracking the maturation periods and counting of strawberries. The tool employs algorithms to generate digital files that provide horticulturists with timely and accessible data on their crops. Key features of the tool include the identification of fruits within video frames, consistent tracking of each identified object throughout the sequence to maintain accuracy, and the extraction of these objects for counting and maturity assessment, among other metrics. This innovation could significantly boost production and establish higher quality standards.

Key Words - Strawberry cultivation, Crop monitoring, Automated tracking, Maturation assessment, Digital agriculture

Introducción

En el contexto de la agricultura moderna, la eficiencia y precisión en el manejo de los cultivos son fundamentales para maximizar la producción y asegurar la calidad del producto. Tradicionalmente, el seguimiento del desarrollo y madurez de los cultivos, como las fresas, ha dependido en gran medida de métodos manuales que son tanto tiempo intensivos como propensos a errores. Frente a este desafío, la tecnología de visión por computadora ofrece una alternativa prometedora, capaz de transformar radicalmente las prácticas agrícolas mediante la automatización del seguimiento y análisis de los cultivos.

Este documento presenta una propuesta para la monitorización de cultivos de fresas utilizando tecnologías de análisis de video. Al integrar técnicas de visión artificial y algoritmos de procesamiento de imágenes, esta herramienta busca superar las limitaciones de los métodos convencionales, proporcionando un sistema automatizado que no solo mejora la eficiencia, sino que también incrementa la precisión de las mediciones de madurez y cantidad de las fresas. La implementación de esta tecnología no sólo optimiza los procesos de inspección y seguimiento, sino que también contribuye a una mejor gestión de los recursos y una planificación más efectiva de las actividades agrícolas.

Este enfoque propone una solución integral que abarca desde la captura de video hasta el análisis detallado de datos, asegurando un monitoreo continuo y confiable esencial para la toma de decisiones informadas en el sector agrícola. En este documento se detallan los desafíos específicos que aborda esta propuesta, las soluciones tecnológicas implementadas y los beneficios esperados de su aplicación.

Índice de figuras

4.1. Cilindro del modelo de color HSV [49].	12
4.2. Características Morfológicas para dos objetos detectados [33].	14
4.3. Centroides Viejos (Morados) y Centroides Nuevos (Amarillos) [33].	15
5.1. Desarrollo Metodológico.	22
5.2. Esquema del Algoritmo de Obtención de Máscaras.	24
5.3. Algoritmo de Identificación, Seguimiento y Extracción.	25
5.4. Esquema del Algoritmo de Cálculo del Estado de Madurez.	26
5.5. Productos del Algoritmo de Obtención de Imágenes.	27
5.6. Imagen Original de Ejemplo.	28
6.1. Imagen Original en Espacio de Color HSV y Clusterizada.	32
6.2. Imagen Original Convertida en Máscara.	33
6.3. Máscara Aplicada a la Imagen Original.	33
6.4. Objetos Detectados Extraídos en la Imagen Original.	35
6.5. Imagen Original con los Objetos Detectados Resaltados.	36
6.6. Objetos Detectados en la Toma Completa.	39

Índice de cuadros

6.1. Centroides de los Clusteres de las Fresas	34
6.2. Muestra de los Datos Registrados	37
6.3. Datos de Madurez	40

Capítulo 1

Problemática

1.1. Formulación de preguntas

¿Cómo es posible hacerle seguimiento a la madurez de fresas, extrayéndolas de imágenes de tomas en movimiento e identificándolas individualmente de cada metraje, en entorno controlado para caracterizarlas, contarlas y caracterizarlas?

1.2. Definición del problema

La medición del porcentaje de madurez de las fresas en la industria agrícola implica una ventaja al determinar la calidad del producto, y en la optimización de los procesos de cultivo y distribución. Sin embargo, los métodos tradicionales basados en la inspección visual y la extracción manual de las fresas resultan bastante laboriosos, lo que limita la eficiencia y certinidad de las mediciones.

La visión artificial es una alternativa al proceso tradicional, formulando nuevos desafíos técnicos para el monitoreo de los frutos. Al identificar las fresas presentes en cada fotograma del metraje; la variabilidad en la apariencia y posición de las fresas, así como la presencia de información no relevante en las imágenes, son consideraciones que el algoritmo tiene que ser capaz de gestionar. Al alimentar el algoritmo con imágenes como archivos individuales que componen el metraje, es necesario mantener una consistencia y no reconocer como diferente el mismo objeto reconocido en imágenes diferentes, así como no registrar como similar objetos diferentes que descritos con parámetros de posición o tamaño similares.

Finalmente, la madurez se puede calcular de una manera similar al método tradicional, a partir de parámetros gráficos, al establecer una relación matemática entre los colores que indican madurez y los que no, que componen la imagen de cada fruto, lo que implica ser capaz de parametrizar con precisión los colores esenciales bajo los cuales se define la madurez e inmadurez.

1.3. Delimitación del problema

El proyecto implementa un dispositivo con una cámara que toma un metraje, pero no almacena la información en un único archivo sino en una carpeta con cada fotograma individual.

Se limita únicamente a fresas y no es válido, bajo los parámetros establecidos, para extrapolarlos con otros frutos ya que eso implica abarcar consideraciones que no se tienen en esta propuesta.

El entorno tiene que ser controlado aunque, por la naturaleza del desarrollo, este aspecto sólo afecta a la detección y no al seguimiento. Es necesario mantener consistencia de los objetos entre la secuencia de imágenes, más que por tamaño y posición, por la posibilidad de que en el procedimiento de detección, puedan dejar existir elementos en imágenes puntuales de la secuencia

Se incluyen tanto fresas maduras como en etapas tempranas de desarrollo, para tener la capacidad monitorear la madurez en todo el proceso de producción, con variaciones en la apariencia, como diferentes tamaños o formas. Se considerarán, asimismo, situaciones en las que las fresas puedan estar agrupadas o superpuestas en la imagen, reflejando condiciones realistas de una banda transportadora donde las fresas pueden estar apiladas o en contacto cercano.

Capítulo 2

Justificación

El proyecto es una propuesta de solución técnica, mediante la aplicación de tecnologías de visión por computadora a través de la segmentación cromática, para superar las limitaciones de los métodos convencionales. Este enfoque permite la identificación y seguimiento automatizado de cada fruto en secuencias de video, adaptándose a la variabilidad de su apariencia causada por factores externos. Una correlación matemática entre los colores y los grados de madurez de las fresas habilita una estimación precisa del estado de madurez, lo que puede significar una mejora en los procesos de control de calidad y gestión agrícola.

En el núcleo de esta propuesta yace la premisa de que la integración de la inteligencia artificial y el análisis de imágenes puede producir un avance disruptivo en la forma en que se monitorea y evalúa la madurez de las fresas a nivel industrial. Este paso intermedio hacia la automatización completa del seguimiento de la madurez no solo aborda una brecha crítica en la precisión y eficiencia operativa, sino que también establece una base para la adaptabilidad y escalabilidad futuras en el sector agrícola. Al reducir la dependencia de la inspección visual humana, que es inherentemente variable y sujeta a errores, el proyecto anticipa una notable mejora en la consistencia de la calidad del producto final. Además, esta transición hacia métodos tecnológicamente avanzados representa una convergencia estratégica de la ciencia de datos con las prácticas agrícolas, marcando un punto de inflexión en cómo la tecnología puede ser aplicada para resolver desafíos tradicionales, elevando el estándar de eficiencia y productividad en la industria.

La implementación de este desarrollo no solo promete optimizar los métodos de determinación de madurez de las fresas, sino que también busca promover una gestión más óptima en la industria agrícola y desarrollar los procesos de cosecha y distribución. Además, el proyecto busca fomentar avances en la investigación agrícola y la tecnología de procesamiento de imágenes, evidenciando su relevancia y potencial impacto positivo en la calidad de los productos y satisfacción de los consumidores.

Capítulo 3

Objetivos

3.1. Objetivo General

Desarrollar un algoritmo para reconocer fresas en imágenes de tomas en movimiento, en entorno controlado, para poder monitorear las mediciones de porcentaje de madurez, por medio de su extracción en cada fotograma e identificación individual.

3.2. Objetivos Específicos

1. Desarrollar un algoritmo que permita iterar por los fotogramas de cada toma, para automáticamente identificar las fresas, generando una máscara y una copia con la máscara aplicada a la imagen inicial.
2. Desarrollar un algoritmo que permita iterar por las máscaras de cada fotograma, para automáticamente marcar las fresas encontradas con un identificador, y hacerles seguimiento, de tal manera, que no se reconozcan como objetos diferentes entre los archivos que componen la toma.
3. Desarrollar un algoritmo que permita iterar por cada fresa identificada en la toma, para automáticamente realizar la segmentación cromática y su posterior determinación de porcentaje de madurez.

Capítulo 4

Marco Teórico

4.1. Reconocimiento de Imágenes y Visión Artificial

La visión artificial es el conjunto de métodos para adquirir, procesar, analizar y entender imágenes digitales, y la extracción de datos de alta dimensionalidad, aquellos que tienen un gran número de dimensiones o características parametrizadas, por medio de la cual producir información numérica o simbólica [3] [27] [47].

La transformación de imágenes, análogo humano del trabajo de la retina, en descripciones del mundo que puedan tener sentido a través de procesos digitales, segregando la información en los campos de la geometría, física, estática y teoría del aprendizaje.

Las fuentes de información pueden tomar varias formas, secuencias de video, vistas desde varias cámaras o estéreo, datos multidimensionales de escáner 3D, nubes de puntos 3D de sensores LiDaR o dispositivos médicos de escaneo. Algunos dominios de la visión artificial incluyen la reconstrucción de escenas, la detección de eventos, el reconocimiento de actividad, estimación de postura en 3D, aprendizaje, indexación, estimación del movimiento, servocontrol visual, donde actuadores son controlados por información física extraída de las imágenes, modelamiento 3D y restauración de imágenes. El desarrollo de este documento abarcará la detección de objetos, seguimiento de videos y reconocimiento de objetos [1].

La detección de objetos es una tecnología computacional que se encarga de la detección de instancias de objetos semánticos de ciertas clases definibles, humanos, edificios, carros, en imágenes digitales o digitalizadas y videos. Conceptualmente, cada clase de objeto tiene características especiales que ayudan a clasificarlo, por ejemplo, los círculos son redondos. Los métodos para la detección de objetos generalmente se divide en aquellos basados en redes neuronales y los que no; mientras que las técnicas neurales son capaces de hacer detección de objetos de extremo a extremo sin definir características específicas, otros métodos requieren aquellas características definidas bajo parámetros computables [2] [5][38].

Más específicamente, para la detección de objetos en movimiento, múltiples fotogramas consecutivos de un video son comparados por varios métodos para determinar si algún objeto se detecta en movimiento. Se puede dividir en cuatro métodos principales, la sustracción del fondo, el flujo óptico, la diferenciación temporal, donde se identifica el objeto en movimiento al aplicar métodos de diferencia pixel a pixel con dos o tres fotogramas consecutivos, y, el relevante en esta

investigación, le diferenciación de fotogramas, que consta de comparar dos imágenes sucesivas para detectar el movimiento [22] [53].

4.1.1. Espacios de Color y el Modelo HSV

Un espacio de color es una organización específica de colores; en combinación con la perfilación de colores soportada por varios tipos de dispositivos físicos, es posible generar representaciones tanto análogas como digitales. Un espacio de color puede ser arbitrario o estructurados con rigor matemático y es una herramienta conceptual útil para entender características adicionales de las imágenes que no pueden ser representadas ‘a simple vista’ [3].

Un modelo de color es un modelo matemático que describe la manera como los colores pueden ser representados como tuplas de números. Existen modelos comerciales como el sistema de color Munsell, el sistema de coincidencia Pantone o el sistema de color natural; de propósito especial como la cromaticidad RG o el espacio de color TSL, usado en detección de rostros; y modelos genéricos como el RGB, en el que vemos las imágenes digitales diariamente, que usa una mezcla aditiva del color que describe qué tipo de luz necesita ser emitida para producir un color entre fuentes roja, verde y azul [45].

Otro de los modelos más importantes son el CMYK, cian, magenta, amarillo y negro, que usa una mezcla sustractiva de colores en procesos de impresión que describe qué tipo de tintas tienen que aplicarse para que la luz reflejada de un sustrato refleje la luz esperada; el HSL, tono, saturación y luminosidad, que es muy similar al modelo HSV, tono, saturación y brillo, cuya principal ventaja es la facilidad de entender una imagen en términos de tono y saturación que de sumas o restas de componentes de color [18] [28] [46].

Modelo de Color HSV

El modelo de color HSV (Hue, Saturation, Value) es una representación de coordenadas cilíndricas de puntos en un modelo de color RGB, reordena su geometría con la intención de ser más intuitivo y perceptualmente relevante que la representación cartesiana o cúbica en la que los colores están en un plano de dos dimensiones. Desarrollado en 1970 para aplicaciones gráficas computacionales, su uso se concentra en detectores de color y edición de imágenes, y no tanto en análisis de imágenes o visión artificial [48].

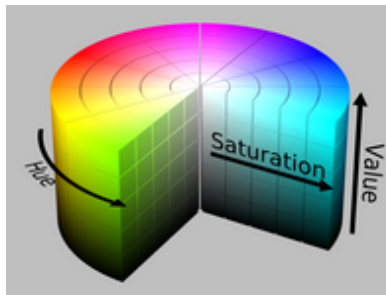


Figura 4.1: Cilindro del modelo de color HSV [49].

En cada cilindro, el ángulo alrededor del eje vertical central corresponde al tono, la distancia desde el eje corresponde a la saturación, y la distancia a través del eje ortogonal a la de la

saturación corresponde al brillo. Este modelo de color es usado en la visión artificial para la detección de características o la segmentación cromática, detección de objetos, reconocimiento de objetos, recuperación de objetos basada en contenido, e imágenes médicas [42].

Aunque dicho espacio de color sirve muy bien para escoger un único color, ignoran mucho de la complejidad de la apariencia del color. Esencialmente, intercambia la relevancia perceptual por rapidez computacional [16].

Conversión de RGB a HSV

La conversión de RGB a HSV [6] [12] se realiza bajo la consideración de que los colores RGB de la imagen de origen deben ser mapeados en valores entre 0 y 1:

$$R, G, B \in [0, 1] \quad (4.1)$$

El componente máximo da el brillo:

$$X_{max} = \max(R, G, B) = V \quad (4.2)$$

El componente mínimo:

$$X_{min} = \min(R, G, B) = V - C \quad (4.3)$$

A partir de los cuales se determina el tono común:

$$H = 0 \Rightarrow C = 0 \quad (4.4)$$

$$H = 60^\circ \cdot \frac{G - B}{C} \text{ mod } 6 \Rightarrow V = R \quad (4.5)$$

$$H = 60^\circ \cdot \frac{B - R}{C} + 2 \Rightarrow V = G \quad (4.6)$$

$$H = 60^\circ \cdot \frac{R - G}{C} + 4 \Rightarrow V = B \quad (4.7)$$

Y diferentes saturaciones:

$$S_V = 0 \Rightarrow V = 0 \quad (4.8)$$

$$S_V = \frac{C}{V} \quad (4.9)$$

Estos principios permiten utilizar el modelo HSV para realizar una segmentación cromática más eficaz, identificando de manera precisa los colores que indican madurez y aquellos que no, lo cual es crucial en la evaluación de la calidad y madurez de las fresas en un entorno controlado.

4.1.2. Enmascaramiento

En las ciencias de la computación, una máscara es un dato usado para operaciones lógicas a nivel de bits. El trazado de recorte es una forma, silueta, cortada en imágenes 2D, donde cualquier cosa adentro de un recorte se mantiene en la imagen resultante de la operación de intersección entre una imagen con información y una máscara, omitiendo la información irrelevante excluida por el trazado de una máscara [15] [43] [52].

El proceso de reconocimiento de objetos en las imágenes se sustenta en el uso en conjunto del cambio de espacio de color de una imagen de entrada RGB a HSV, donde se aprovecha la característica de dicho espacio de color de darle prioridad a los colores que a los detalles, permitiendo generar un contraste alto entre las frutas y el fondo; y la generación de máscaras de trazados por recorte, donde se puede realizar una operación computacional para convertir los valores en el rango de tonos definidos para recortar las fresas y omitir el fondo [13] [39].

4.2. Seguimiento de Objetos Fotograma por Fotograma

4.2.1. Algoritmos de Distancia Euclidiana

Los algoritmos de seguimiento de distancia euclidiana son algoritmos de seguimiento de objetos simple que permite aplicar un identificador único a cada objeto detectado, lo que permite contar objetos individuales en videos y sus secuencias de fotogramas. Los algoritmos de distancia euclidiana dependen del cálculo de la distancia entre características morfológicas (Fig. 4.2) de los objetos identificados entre un fotograma y el otro, sea el siguiente o el anterior [3] [4].

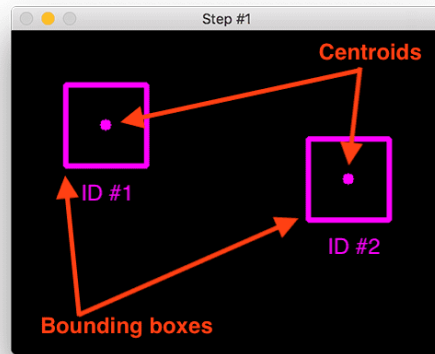


Figura 4.2: Características Morfológicas para dos objetos detectados [33].

Estos algoritmos se alimentan de conjuntos de coordenadas de centroides o cajas delimitantes para cada objeto y en cada fotograma, donde en la primera imagen de cada secuencia se le asigna un identificador a cada objeto. Posteriormente, se computa la distancia euclidiana entre los centroides originales y los nuevos (Fig. 4.3).

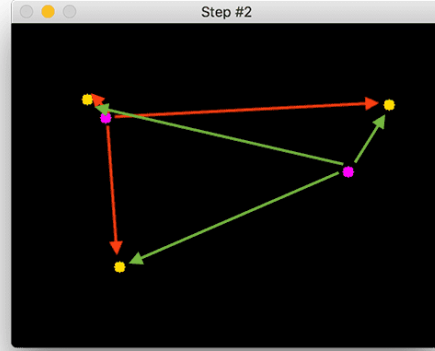


Figura 4.3: Centroides Viejos (Morados) y Centroides Nuevos (Amarillos) [33].

La primera suposición del algoritmo es que un objeto se va potencialmente a mover entre fotogramas subsecuentes, pero la distancia mínima calculada entre las características de los objetos de los fotogramas continuos para tener el mismo identificador es la menor al comparar las del objeto final con todos los de la imagen anterior [53] [54].

Para el registro de nuevos objetos, se añaden nuevos identificadores a la lista de los existentes, posterior a asignar los existentes a sus correspondientes objetos; así mismo, el algoritmo debe ser capaz de manejar cuando un objeto se ha perdido, desaparecido o salido del campo de la vista, caso que sucede cuando un identificador no se puede relacionar con un objeto existente, normalmente porque se sale del rango definido para la distancia mínima.

Dos de los principales tipos de rastreadores por distancia euclidiana son el de centroide, que hace los cálculos a partir del objeto central de cada objeto, a los cuales se les aplica los cálculos de distancias; el de caja delimitadora, que hace el seguimiento con el rectángulo completo que contiene la totalidad de los objetos [21] [22].

4.2.2. Intersección Sobre Unión

También conocido como índice Jaccard, es una estadística usada para medir la similitud y diversidad de conjuntos de muestras. Mide el coeficiente de similaridad entre conjuntos de muestras finitas y se define como el tamaño de la intersección dividido por el tamaño de la unión de las muestras contenidas en los conjuntos [10] [51].

$$J(A, B) = \frac{A \cap B}{A \cup B} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|} \quad (4.10)$$

Para determinar la intersección sobre unión con el uso de máscaras, los conjuntos de muestras se definen por las máscaras de cada fotograma, donde la silueta correspondiente a las fresas son los conjuntos de muestras de interés, y el cálculo de intersección sobre unión se aplica calculando la sobreposición de las siluetas entre los fotogramas consecuentes [32].

4.3. Maduración de las Fresas y Segmentación Cromática

La determinación visual de la madurez de las fresas se basa en la observación de varias características como los cambio de color de verde a tonos rojizos, indicando un mayor grado de madurez. También se considera la textura suave y carnosa de las fresas maduras, en contraste con la firmeza de las fresas inmaduras. Además, se evalúa el sabor dulce y el aroma intenso que suelen estar asociados con las fresas maduras [50].

La evaluación de la madurez de las fresas puede beneficiarse del análisis de visión artificial, particularmente mediante la segmentación cromática. Las técnicas como la clusterización y el análisis de espacio de color HSV pueden facilitar la identificación de los estados de madurez al segmentar y clasificar las áreas de color de las imágenes de las fresas [2] [38].

La evaluación visual suele considerar igualmente el tamaño y la forma de las fresas, buscando uno más grande y una forma redondeada. Si bien esta evaluación es subjetiva y depende de la experiencia del observador, proporciona una indicación superficial de la madurez de las fresas y puede ser complementada con técnicas analíticas de mayor rigor [14] [36].

4.3.1. Clusterización

O análisis de clústeres es la tarea de agrupar objetos en conjuntos de tal manera que aquellos en el mismo grupo son los más similares entre sí que con otros elementos de la muestra. Su tarea es el análisis de datos tanto exploratorio como estadístico; usado en bastantes campos, como el reconocimiento de patrones, análisis de imágenes, recuperación de información, bioinformática, comprensión de datos, gráficos computacionales y aprendizaje automático [20] [23] [44].

El algoritmo de k-means es una técnica iterativa que se usa para partir una imagen en clústeres k. Se determina un centroide del clusteres, que puede ser escogido aleatoriamente o basándose en cualquier método heurístico, práctico; se asigna a cada pixel en la imagen el valor que minimiza la distancia entre dicho pixel y el centroide más cercano; se computan los valores de los pixeles de tal manera que se obtiene una imagen con sus valores promediados a los de los más cercanos, o en otras palabras, imágenes con una cantidad finita y definida de colores [24].

Tanto para la generación de máscaras como para el cálculo de madurez, se usa la clusterización en la medida que permite la simplificación de las imágenes para su procesamiento. En el primer caso, ejecutar una clusterización sobre la imagen HSV permite que a partir de las dos tuplas con los valores con los que fueron promediados los pixeles de las imágenes, dos únicos valores conocidos que permiten una binarización gráfica directa. En el segundo caso, clusterizar la silueta de la fresa permite dividirla en dos tonos diferentes con los cuales se puede hallar un coeficiente entre el área del color maduro, con respecto al área total de la silueta [7] [55].

4.4. Estado del Arte

El cálculo de la madurez de las frutas por medio de la visión artificial ha tenido recientes avances con notables casos de estudio. En la india, se desarrolló un sistema para automatizar el gradiente de los mangos con técnicas de aprendizaje profundo llamadas redes neuronales convolucionales. Este sistema alcanza una precisión del 98 % en la detección de defectos en la superficie de los mangos, incrementando significativamente la calidad del proceso de cálculo de

gradientes para mercados de exportación [31] [35].

Por medio del uso de imágenes obtenidas por drones, un sistema basado en visión artificial se implementó para monitorear los cultivos de arándanos. Se incluyó la creación de conjuntos de datos CRAID, los cuales contienen las imágenes de varias variedades de arándanos recolectados en diferentes periodos de cosecha para apoyar la evaluación de la salud de las bayas. Dicho enfoque involucró calibración fotométrica y segmentación semántica débilmente supervisada para localizar con precisión los arándanos en imágenes aéreas [8].

En un estudio conducido en Irán, un sistema de visión por computadora emparejado con técnicas de aprendizaje profundo, específicamente los modelos Inception-V3 y ResNet-50, usados para clasificar los frutos de espino entre las categorías inmaduros, maduros y pasados de maduros. Ese método demostró gran asertividad y efectividad al incrementar la comerciabilidad y reducir el desperdicio al mejorar los métodos de gradiente tradicional de las frutas [9].

En Australia, se implementó un sistema basado en visión artificial para clasificar cerezas según su madurez y calidad. Utilizando técnicas de procesamiento de imágenes y algoritmos de aprendizaje automático, este sistema logró aumentar la eficiencia en la selección y empaquetado de cerezas, reduciendo significativamente la cantidad de fruta desechada por errores humanos [41].

Investigadores en Brasil se desarrolló un modelo utilizando redes neuronales convolucionales para estimar la madurez de bananas a partir de imágenes digitales. Este modelo proporcionó una herramienta precisa y no destructiva para la industria bananera, facilitando la distribución más efectiva de la fruta y optimizando los tiempos de venta [11].

En España, se implementó un sistema de visión por computadora para monitorizar la calidad de las uvas durante el proceso de vinificación. Utilizando imágenes multiespectrales y análisis de imágenes, este sistema permite identificar precozmente problemas de calidad y enfermedades en las uvas, asegurando una mejor calidad del vino final [25].

También en España se desarrolló un sistema de visión por computadora para automatizar el conteo de fresas en invernaderos. El sistema utiliza cámaras de alta resolución y algoritmos de procesamiento de imágenes para identificar y contar de manera precisa las fresas maduras en los campos de cultivo. Esta tecnología ha demostrado ser una herramienta valiosa para los agricultores al permitirles monitorear el crecimiento de los cultivos y planificar la cosecha de manera más eficiente [17].

Un estudio realizado por un equipo de investigadores en Japón, desarrolló un sistema de clasificación automatizado para fresas basado en la visión por computadora. El sistema utiliza una combinación de imágenes multiespectrales y algoritmos de aprendizaje profundo para evaluar la calidad de las fresas, incluyendo su madurez, color y posibles defectos. La precisión del sistema fue evaluada y demostró ser altamente eficaz, ofreciendo un método rápido y no invasivo para la clasificación de fresas en ambientes de producción agrícola [40].

En México, se implementó un sistema de análisis de imágenes basado en visión por computadora para optimizar el riego en cultivos de fresas. El sistema utiliza cámaras instaladas en los campos de cultivo para capturar imágenes de las plantas de fresa y analizar su estado de hidratación. Utilizando algoritmos de procesamiento de imágenes y aprendizaje automático, el

sistema determina cuándo y cuánto regar las plantas para maximizar el rendimiento del cultivo y minimizar el consumo de agua [19].

En Estados Unidos, se desarrolló un sistema de detección temprana de enfermedades en cultivos de fresas utilizando imágenes aéreas y análisis de visión por computadora. El sistema utiliza drones equipados con cámaras multispectrales para capturar imágenes de los campos de cultivo y analizar la salud de las plantas de fresa. Al identificar signos tempranos de enfermedades, los agricultores pueden tomar medidas preventivas para proteger sus cultivos y minimizar pérdidas en la producción [37].

El estado del arte en el cálculo de la madurez de frutas mediante visión artificial ha experimentado avances significativos en diversos países. Desde sistemas automatizados de gradiente de mangos en India hasta la clasificación de cerezas en Australia, y desde la monitorización de la calidad de las uvas en España hasta la detección temprana de enfermedades en cultivos de fresas en Estados Unidos, estos casos de estudio demuestran la efectividad de la visión por computadora en la optimización de procesos agrícolas. Con el uso de técnicas como redes neuronales convolucionales y análisis de imágenes multispectrales, estos sistemas ofrecen herramientas precisas y no destructivas que mejoran la calidad, comercialización y eficiencia en la producción de frutas a nivel mundial.

Capítulo 5

Método Experimental y de Investigación

5.1. Diseño Metodológico

5.1.1. Tipo de Investigación

El proyecto utiliza una investigación de un enfoque mixto, que combina tanto elementos cuantitativos como cualitativos. La investigación aplicada busca desarrollar y aplicar conocimientos prácticos para resolver un problema concreto. En este caso, el objetivo es poder realizar una estimación de la madurez de las fresas utilizando visión artificial, permitiendo contarlas y caracterizarlas individualmente.

El enfoque cuantitativo se emplea para recopilar y analizar datos numéricos relacionados con las características morfológicas de los objetos reconocidos, los identificadores y conteo de las fresas y la segmentación cromática para calcular los porcentajes de maduración de las fresas. Por otro lado, el enfoque cualitativo se emplea para comprender y analizar aspectos subjetivos relacionados con la apariencia y otros atributos de las fresas, a través de observaciones físicas.

Esta combinación de enfoques cuantitativos y cualitativos permite obtener una percepción detallada de la madurez de las fresas, con la convergencia tanto de aspectos objetivos como subjetivos. Además, facilita la generación de conclusiones y recomendaciones basadas en evidencia, que puedan ser aplicadas en entornos controlados y en la industria agrícola.

5.1.2. Recolección y Procesamiento de Datos

1. Análisis de requisitos y definición del problema: Se realizará un análisis de los requisitos y necesidades del proyecto, se identificarán los objetivos específicos y los recursos disponibles. Además, se estudiará en profundidad el problema de la medición de la madurez de las fresas detectadas en videos desde los fotogramas que los componen, sobre la consistencia en la identificación de objetos a lo largo de una secuencia de imágenes y segmentación cromática.
2. Diseño del sistema de visión artificial: Se llevará a cabo el diseño del sistema de visión artificial. Se pondrán a prueba, adaptarán y seleccionarán las técnicas más adecuadas, como cambios de espacio de color, enmascaramiento, métodos de seguimiento de objetos y segmentación de las fresas. Se establecerán los parámetros óptimos para el reconocimiento

y posterior extracción de los frutos encontrados, teniendo en cuenta la variabilidad en la apariencia y posición de los frutos.

3. Implementación y pruebas del sistema: Una vez completado el diseño, se procederá a la implementación de los algoritmos en secuencias de prueba. Estas pruebas permitirán evaluar la efectividad y precisión del sistema en la detección y seguimiento de las fresas, y en la estimación del grado de madurez.
4. Evaluación de resultados y validación: En esta fase, se evaluarán los resultados obtenidos por el sistema de visión artificial. Se compararán las mediciones de madurez realizadas por el sistema con las mediciones manuales de referencia. Se realizarán análisis estadísticos para determinar la precisión y confiabilidad del sistema en la estimación de la madurez de las fresas. Además, se llevará a cabo una validación para evaluar la calidad y rendimiento de los algoritmos.
5. Documentación y difusión de resultados: Finalmente, se documentarán todos los aspectos del proyecto, desde el diseño hasta los resultados y conclusiones. Se elaborará un informe técnico detallado que describa la metodología utilizada, los resultados obtenidos y las implicaciones prácticas del sistema desarrollado.

5.1.3. Análisis de Datos

1. Extracción de características: Se aplican técnicas de procesamiento de imágenes para extraer características específicas de las fresas. Esto implica la utilización de algoritmos para aislar las regiones de interés en cada imagen para su procesamiento. Se busca registrar medidas cuantitativas para describir los objetos por su morfología, sus colores y adicionalmente el elemento identificador.
2. Cálculos y mediciones: Utilizando las medida extraídas se realizan cálculos para determinar los rangos de valores que determinan las máscaras, los parámetros de funcionamiento de los algoritmos de seguimiento, distancias euclidianas e intersección sobre unión y los grados de madurez de cada fruta. Se establecen umbrales de color que indican el nivel de madurez de las fresas en función de tonos específicos. Se emplean algoritmos de para asignar una etiquetas y resaltar en cada fruta las características obtenidas.
3. Análisis y visualización: Los resultados de los cálculos y mediciones serán analizados estadísticamente para evaluar la relación entre las características extraídas y el grado de madurez de las fresas. Se utilizarán herramientas y técnicas estadísticas para identificar patrones, correlaciones y tendencias relevantes. Además, se generarán visualizaciones gráficas, como histogramas o gráficos de dispersión, que permitan representar de manera clara y concisa la distribución de la madurez de las fresas en función de las características analizadas. Esto facilitará la interpretación de los resultados y la comunicación efectiva de los hallazgos.

5.2. Etapas y Actividades

5.2.1. Etapa de Preparación y Presentación de la Propuesta.

1. Identificación y definición del problema a tratar.
2. Preparación de la propuesta de investigación, incluyendo los objetivos, la delimitación del problema, el tipo de investigación y el marco teórico.

3. Presentación y discusión de la propuesta.

5.2.2. Etapa de Investigación Preliminar.

1. Revisión de literatura y trabajos anteriores relacionados con la identificación de objetos.
2. Revisión de literatura y trabajos anteriores relacionados con el seguimiento de objetos.
3. Revisión de literatura y trabajos anteriores relacionados con métodos de segmentación cromática para determinar la madurez de las fresas.

5.2.3. Etapa de Diseño del Algoritmo.

1. Diseño del algoritmo para iterar los archivos de la carpeta con los archivos de una determinada toma, que generan una copia en espacio de color HSV, una con una máscara donde se extraen los frutos y una al aplicar la máscara en la imagen original.
2. Diseño del algoritmo para iterar los archivos de la carpeta con las máscaras halladas, sobre los cuales hacer el seguimiento a las fresas, manteniendo la consistencia de objetos entre imágenes y guardando cada objeto identificado en carpetas para cada fotograma.
3. Diseño del algoritmo para iterar entre los archivos fresas detectadas para, con segmentación cromática, calcular los correspondientes niveles de madurez.

5.2.4. Etapa de Implementación, Pruebas y Ajustes.

1. Pruebas iniciales y correcciones de errores en un entorno controlado con un número limitado de fresas.
2. Mejoras del algoritmo con base en los resultados de las pruebas.
3. Realización de pruebas a mayor escala con un conjunto de datos diferente y más grande.
4. Ajustes y mejoras basados en los resultados de las nuevas pruebas.
5. Comprobación de la robustez del algoritmo, por medio de métricas, con énfasis en la asertividad de la predicción, y variaciones en la apariencia y la agrupación o superposición de las fresas.

5.2.5. Etapa de Documentación, Preparación de Informes y Divulgación.

1. Documentación detallada del proceso de investigación y desarrollo, incluyendo problemas encontrados y cómo se resolvieron.
2. Preparación del informe final y presentación de los resultados.
3. Divulgación de los resultados y hallazgos a través de diferentes canales.

5.3. Recursos y Métodos

5.3.1. Materiales y Presupuesto

Hardware

1. Computador: \$5 000 000.

Software

1. Python 3.11, para desarrollo de algoritmos, procesamiento de imágenes y análisis de datos.

Otros Materiales

1. Fresas en diferentes estados de maduración: \$100 000.

5.3.2. Institucionales

Hardware

1. Sistema de captura de imágenes.
2. Entorno de trabajo e iluminación.

Software

1. OneDrive, para control de archivos y versiones.

5.4. Base Teórica y Simulaciones

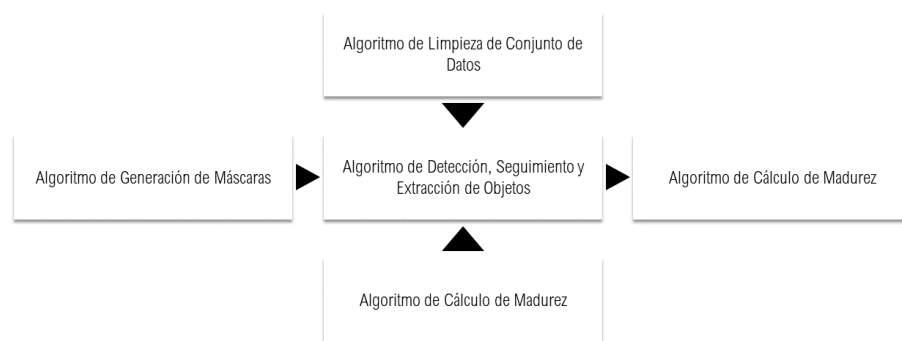


Figura 5.1: Desarrollo Metodológico.

5.4.1. Algoritmo de Obtención de Máscaras

Inicialización

Este proceso del algoritmo lo constituye la definición de los directorios de entrada y de salida, incluyendo directorios para clasificar los archivos de salida con respecto a atributos relevantes en la ejecución del programa. Para la preparación del entorno se leen los nombres de los archivos contenidos en el directorio de entrada y se ordenan en una lista a partir de su índice numérico.

Procesamiento

A partir de la lista de imágenes que van a ser procesadas por el código, se entra en un ciclo en el cuál, para cada imagen, se hace una serie de procesos en los que se generan cuatro archivos de salida para cada uno en la entrada.

En la secuencia repetida dentro del bucle, se carga el archivo al sistema; se realiza una conversión de espacio de color a HSV a la cual se le hace una clusterización k-means para simplificar la imagen, promediándola para dos valores de colores puntuales a partir de los cuales se genera una máscara, el valor de color que corresponde al de la silueta de las fresas se convierte en blanco y el resto en negro.

Al aplicar una operación de intersección entre el fotograma original y la máscara se produce una secuencia similar a la original, donde se resalta la información relevante y se omite por completo la información irrelevante del fondo.

Salida

Los archivos que genera este algoritmo, por lo tanto, son la máscara, la copia del fotograma con la máscara aplicada y una copia de la imagen en espacio de color HSV para corroborar los colores del rango.

Consta de mensajes de ejecución que indican cuando una imagen está siendo procesada y cuando los archivos de salida generados son guardados, así como los mensajes de error que notifican cuando un archivo no se puede leer o guardar. Finalmente se genera un mensaje que indica el tiempo de ejecución del código [26] [29] [30] [34].

5.4.2. Algoritmo de Identificación, Seguimiento y Extracción de Objetos

Inicialización

Este proceso del algoritmo lo constituye la definición de los directorios de entrada y de salida, incluyendo directorios para clasificar los archivos de salida si corresponden a los objetos extraídos de cada fotograma, o las máscaras completas donde se resaltan los objetos con su identificador, así como un archivo CSV donde se registra junto con sus parámetros morfológicos como el centroide, la caja delimitante y el área de la máscara.

Para la preparación del entorno, primero se ejecuta el algoritmo para la limpieza del conjunto de datos, donde se eliminan los archivos sin información gráfica del resto de imágenes; de segundo, se escoge el tipo de algoritmo de seguimiento entre las clases codificadas y con los que se va a

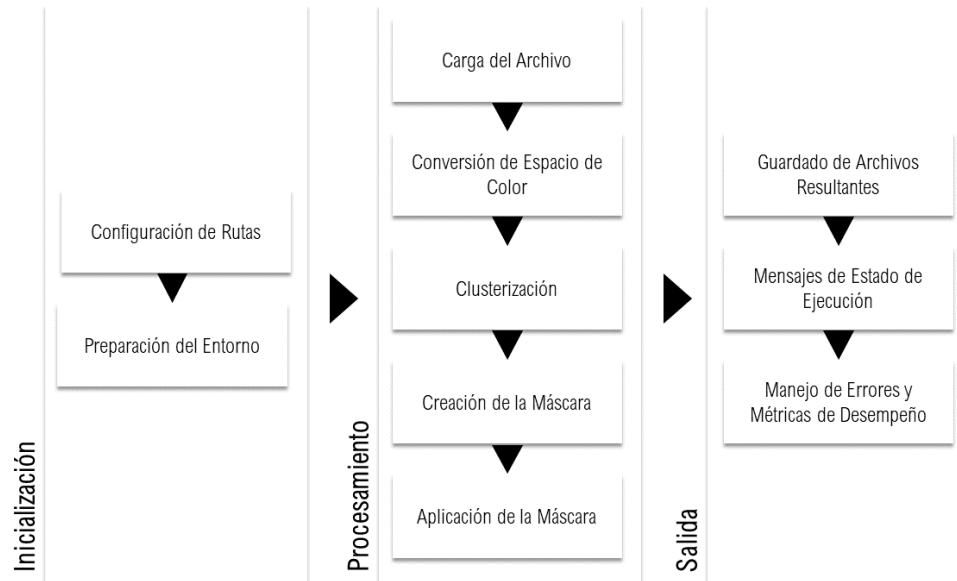


Figura 5.2: Esquema del Algoritmo de Obtención de Máscaras.

llevar a cabo el procedimiento; y finalmente se leen los nombres de los archivos contenidos en el directorio de entrada y se ordenan en una lista a partir de su índice numérico.

Procesamiento

A partir de la lista de imágenes que van a ser procesadas por el código, se entra en un ciclo en el cuál, para cada imagen, se hace una serie de procesos en los que se generan tanto un conjunto de archivos con recortes de las fresas que componen cada fotograma que es exitosamente procesado y una copia de la máscara con dichos objetos identificados y resaltados.

En la secuencia repetida dentro del bucle, se carga el archivo al sistema; se hace una conversión de espacio de color, esta vez a escala de grises debido a que el método requiere que la matriz así esté definida para identificar los contornos, donde la naturaleza de las imágenes como máscaras facilitan su detección. Después se realiza el seguimiento, en el primer fotograma se asignan los primeros identificadores y a partir de ellos se gestiona las designaciones posteriores.

Salida

Los archivos que genera este algoritmo, por lo tanto, son los recortes de cada objeto que compone un fotograma en una carpeta que lo identifica, y una copia de la máscara con la cual se computaron los contornos, donde se resalta el objeto con su respectiva caja delimitante y el identificador que le corresponde.

Consta de mensajes de ejecución que indican cuando una imagen está siendo procesada y cuando los archivos de salida generados son guardados, así como los mensajes de error que notifican cuando un archivo no se puede leer o guardar. Finalmente se genera un mensaje que indica el tiempo de ejecución del código [26] [29] [30] [34].

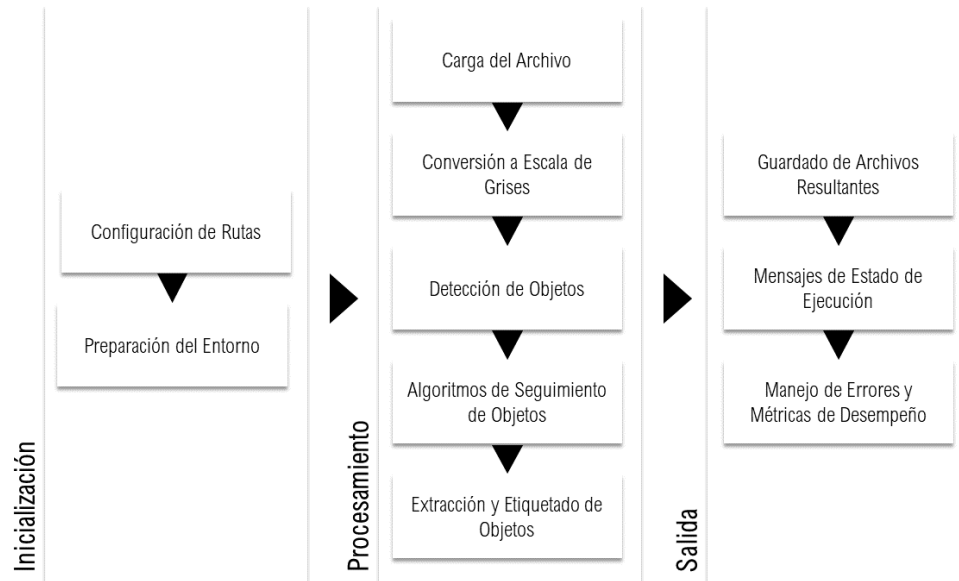


Figura 5.3: Algoritmo de Identificación, Seguimiento y Extracción.

5.4.3. Algoritmo de Cálculo del estado de Madurez

Inicialización

Este proceso del algoritmo lo constituye la definición de los directorios de entrada y de salida, incluyendo directorios para clasificar los archivos de salida que incluyen la carpeta final del proyecto, con un recorte de cada fresa identificada en la ejecución y escogida a partir de la máscara con mayor área correspondiente para un identificador, así como un archivo CSV similar al del algoritmo anterior con los datos registrados exclusivamente para el conjunto final de objetos.

Procesamiento

Después de todo el proceso de inicialización, el algoritmo se remite al archivo CSV creado en el código anterior y que contiene los datos morfológicos registrados para cada objeto identificado en cada fotograma; en este caso son principalmente relevantes el identificador y el área de la máscara.

Se filtra entre los objetos con el mismo identificador, el nombre del archivo donde se encuentra la imagen que haya generado una máscara con un área mayor, y, a partir de esa lista, se entra en un ciclo de procesamiento. A cada objeto de la lista se le realiza un proceso de segmentación cromática similar al del primer algoritmo, para simplificar la silueta de la fresa en dos tonalidades con las cual, posteriormente, se calcula el coeficiente de madurez entre el área definida como madura y el área total de la fresa.

Salida

Los archivos que genera este algoritmo, por lo tanto, son un recorte de las fresas que cumplen el criterio de selección, recorte al cuál se le aplica la máscara correspondiente a dicha silueta, con el propósito de aislar la imagen de la fresa, y un archivo CSV con la información correspondiente a dichos objetos representativos.

Consta de mensajes de ejecución que indican cuando una registro está siendo procesado y cuando los archivos de salida generados son guardados, así como los mensajes de error que notifican cuando un archivo no se puede leer o guardar. Finalmente se genera un mensaje que indica el tiempo de ejecución del código [26] [29] [30] [34].

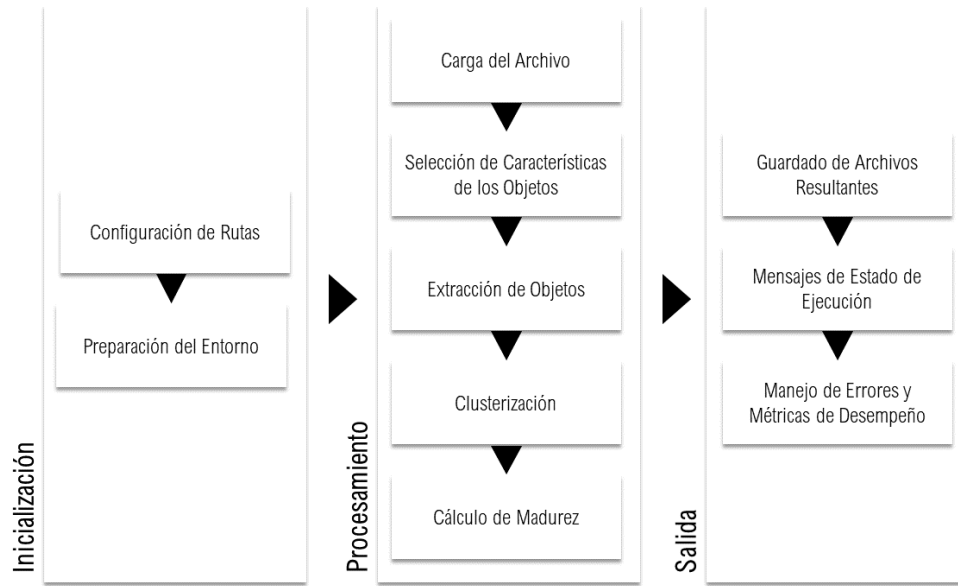


Figura 5.4: Esquema del Algoritmo de Cálculo del Estado de Madurez.

5.4.4. Algoritmos de Seguimiento

El archivo de algoritmos de seguimiento se compone de tres clases compuestas cada una de dos funciones, la típica función `init` que inicializa la clase junto con las variables que se utilizan en la función `update`, que se encarga de actualizar los identificadores con el resultado de su ejecución.

La clase de centroides lee los dichos puntos para los objetos identificados en el fotograma que está siendo procesado y comienza a asignar identificadores en la primera imagen desde el uno en adelante; cuando el valor de la distancia entre los puntos de interés de los objetos cumple con las condiciones de ser la distancia menor con respecto a los demás centroides, y se encuentra dentro de un rango definido por criterio de diseño, se mantiene la consistencia con el identificador; para todos los otros casos se gestiona como objeto nuevo, u objeto removido.

Para la clase de cajas delimitantes el proceso es muy similar, la diferencia radica en que las distancias se calculan no desde los centroides sino desde los vértices de las cajas delimitantes, distancias que tienen que cumplir también con las condiciones descritas.

La clase de intersección sobre unión la componen las mismas clases, tienen parámetros de validación similares donde el resultado de la intersección debe existir dentro de un rango para poder ser identificado como el mismo objeto, y además incluye una función exclusiva a esta clase encargada únicamente del cálculo de la intersección sobre unión.

5.4.5. Algoritmo de Limpieza de Conjuntos de Datos

Este último algoritmo, siendo el más simple de todos, consta de una clase donde se invocan la definición de variables, la función de limpieza que recorre todos los archivos de la carpeta de entrada y que al detectar un archivo vacío, completamente en negro, lo remueve del directorio, y un método estático equivalente a la función que se usa en los algoritmos principales para la organización de los archivos en una lista a partir de su índice.

Se define el método estático porque es el procedimiento análogo en la programación de clases a definir alguna función en un código orientado a objetos. La función identifica la parte numérica del nombre del archivo que se usa como llave en el método sorted.

5.5. Detalles Experimentales

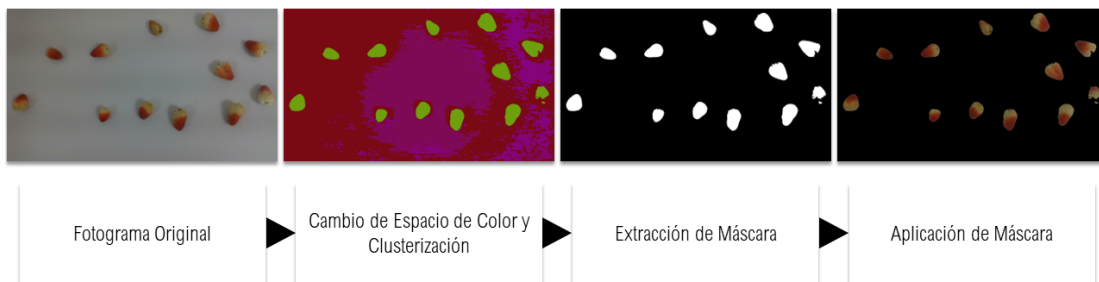


Figura 5.5: Productos del Algoritmo de Obtención de Imágenes.

5.5.1. Algoritmo de Obtención de Máscaras

Tiempo de procesamiento: 67.281 segundos

A partir de cada una de las imágenes que componen las secuencias de las tomas es necesario obtener una máscara de tal manera que se simplifiquen y permitan un procesamiento menos demandante para la detección únicamente de los elementos que necesitan ser evaluados.



Figura 5.6: Imagen Original de Ejemplo.

La primera consideración de diseño viene en el momento en el que se procesa una imagen de la secuencia original, no se le puede hacer una simplificación directamente; por el color del fondo, se da el caso de que el sistema lo confunde con sectores inmaduros, amarillosos de la fresa y tiene es sensible a los cambios de luz. Por lo tanto, se propone un cambio de espacio de color de RGB a HSV, el cual demostró en pruebas que, independientemente de la iluminación, puede generar un contraste bastante alto entre las dos partes interesadas de leer.

Así, se transforma una imagen donde predominan los tonos rojizos, amarillos, blancos, las fresas y el fondo, y en bastante menor medida verdosos, de algunas semillas y tallos, y negros, con sombras y tonos resultantes de la interacción de la cámara y los objetos con la iluminación, a una donde la parte correspondiente a la silueta de la fresa es de tonos verdes, y de todo el resto, tonos rojizos.

Desde esa imagen con únicamente dos colores bastante contrastantes, se propone realizar una clusterización de la nueva imagen; el proceso de clusterización convierte el área correspondiente a la de las fresas en una mancha uniformemente coloreada del tono de verde predominante en todas las fresas, igual que el del fondo, que adquiere en toda su extensión el tono rojizo mayoritario.

Para el siguiente paso es necesario identificar los valores exactos de dichos tonos representativos, en un algoritmo se generó en la imagen en espacio de color HSV, una tabla de convenciones con el valor de los centroides, los valores numéricos representativos del tono, precedido por un pequeño recuadro de aquel color. Con los valores de cada tono diferente de verde identificados a través de toda una secuencia de imágenes, se puede estimar un rango a partir del cual generar una máscara.

La información en este punto nos permite hacer el siguiente proceso de simplificación, en el cual se convierten todos los píxeles con valores en el rango de verdes en blanco absoluto y el espacio remanente completamente en negro. El resultado es una imagen binaria, una máscara, un tipo de imagen con la cual se puede hacer, por ejemplo, el producto matemático con la imagen original.

El producto de un píxel de color con uno blanco es equivalente a la propiedad de identidad de la multiplicación, con uno negro, a la multiplicación con el cero; en otras palabras, el área negra de la máscara se ignora en la imagen original dando como resultado una imagen donde sólo se distingue el área de interés de la imagen inicial, las fresas.

En resumen, el algoritmo de producción de máscaras genera tres productos que se archivan en su correspondiente carpeta de resultados, los dos primeros son la imagen en espacio de color HSV y la imagen donde es aplicada la máscara al fotograma original, ambas proporcionan al usuario material para análisis de segunda mano, para comprobar calibrar el sistema como los rangos para definir las máscaras, entre otros.

5.5.2. Algoritmo de Identificación, Seguimiento y Extracción de Objetos

El producto remanente es la imagen binaria obtenida para cada fotograma de la secuencia, es equivalente a tener la misma toma completamente en blanco y negro. Alimentar el algoritmo de detección y seguimiento de objetos con imágenes binarias es mucho más versátil que con imágenes complejas debido a la capacidad de procesamiento limitada de la máquina.

El proceso de detección de las imágenes se realiza de la siguiente manera, una imagen es cargada al sistema que ejecuta un método de la librería de open CV definido específicamente para identificar contornos; los contornos se determinan al analizar una imagen como un mapa en dos dimensiones de señales sobrepuestas, en los puntos en los que se detecta el equivalente a una discontinuidad en la señal se termina estableciendo como un posible contorno.

Para lo cual, el mismo algoritmo requiere que la imagen obligatoriamente esté en escala de grises y es una ventaja hacerlo con las máscaras; si se usara el fotograma original en escala de grises, la naturaleza de la imagen puede llegar a detectar elementos adicionales que tienen un contorno propio pero cuya información no es relevante para esta aplicación, en la máscara, tenemos únicamente la información esencial, es como asegurar la predicción del sistema.

Ya con los contornos encontrados para una sola imagen, procede introducir ese procedimiento en un ciclo similar al del primer algoritmo para recorrer todas las imágenes de la secuencia y extraer, aún sin una relación de continuidad, los datos descriptivos de toda la toma. A partir de esa información, se hace proponen tres mecanismos para hacer el seguimiento de los objetos y establecer la continuidad y consistencia de cada detección.

Centroid Tracker

Tiempo de procesamiento: 23.141 segundos

Este es el método menos complejo de todos, entre los datos morfológicos interpretados para cada detección, uno es el centroide, el punto central de un elemento definido por la detección de sus contornos. A la hora de tener una colección de centroides, y sabiendo a que imagen específicamente corresponden, el método de distancia euclidiana por centroides, escoge un centroide de la imagen en revisión y evalúa su distancia con respecto a los centroides registrados de los objetos de imágenes anteriores, hace una resta, y organiza jerárquicamente de menor a mayor las distancias calculadas; si una distancia se halla definida dentro de un rango establecido por el usuario, es mayor la probabilidad de que sea el mismo objeto y por lo tanto se le designa el mismo identificador.

Bounding Box Tracker

Tiempo de procesamiento: 23.712 segundos

Por otro lado, este método es muy similar al anterior, pero ligeramente más robusto en la medida en la que el parámetro por el cual se establece la similaridad esta soportado por más puntos morfológicos. La distancia euclidiana por cajas delimitadoras no hace la operación aritmética desde un único valor definido sino toma en consideración los vértices de la caja, por lo cual, ofrece más información para contrastar un elemento con aquellos otros que pertenezcan al rango en el que se establece que un objeto puede ser el mismo en otra imagen.

Intersection Over Union Tracker

Tiempo de procesamiento: 21.882 segundos

La detección sobre unión, por otro lado, aunque su naturaleza también radica esencialmente en una operación aritmética, lo que hace no es calcular distancias sino áreas de solapamiento, por lo tanto el sistema al identificar un contorno hace una relación entre un área de referencia y las siluetas de la imagen siguiente, entre mayor sea el resultado, más se sobreponen y es más probable que corresponda al mismo elemento con cierto factor de movimiento.

Finalmente el producto de este algoritmo son dos carpetas de salida, una en la que se almacenan las máscaras de todas las secuencias con cada objeto detectado resaltado; una en la que se almacenan carpetas, cada una con el número del fotograma al que corresponden, y en las que se almacena el recorte de la máscara de cada objeto identificado; y finalmente un archivo csv en el que se encuentran listados todos parámetros morfológicos extraídos para cada elemento en cada imagen de toda la secuencia.

5.5.3. Algoritmo de Cálculo del estado de Madurez

Para el cálculo de la madurez de las fresas, el sistema está diseñado para ya no interpretar información gráfica sino en texto plano. Todos los datos morfológicos que fueron registrados en el archivo csv, datos en texto principalmente numéricos, permiten realizar operaciones con la información más esencial y haciendo bastante menos demandante los cálculos para el propósito.

El flujo comienza evaluando el archivo de datos csv en donde adicional a los datos morfológicos se encuentra registrado el identificador con el que se ha marcado cada objeto. El sistema coge todos los registros que comparten el mismo identificador obteniendo como resultado una lista con todos los fotogramas en los que se encuentra dicho objeto junto a sus propiedades gráficas, compara las áreas registradas de las máscaras y finalmente escoge como el objeto representativo

aquel con el área mayor entre todas las imágenes en las que se encuentre.

El criterio para escoger ese elemento representativo se soporta en la premisa que los elementos con un mayor área, ofrecen más información para poder realizar una segmentación cromática más precisa, sin embargo es necesario recalcar que la precisión en este caso es un tema de aproximación y no está ligada a un parámetro infalible. Cuando se obtiene al final una lista con los elementos representativos en la secuencia de cada fruto identificado, a partir de la cual, se identifica el elemento en la secuencia de imágenes y se extrae para como un archivo individual.

A ese archivo se le hace un procedimiento aritmético sencillo, donde se clusteriza la imagen simplificándola en dos colores, normalmente cada color corresponde uno al área amarillenta de la fresa, el área inmadura, y el área roja; el área roja de la fresa en relación con el área total, es la aproximación que se utiliza para estimar finalmente el valor aproximado de madurez de los frutos.

Los productos resultantes de este algoritmo son dos, una carpeta con las imágenes de cada elemento detectado, sin duplicados entre fotogramas, manteniendo la consistencia de los objetos, y un archivo csv, similar al del algoritmo anterior, pero cuya información se resume únicamente a cada fruto individual por toma, los frutos representativos.

Capítulo 6

Resultados

6.1. Resultados del Algoritmo de Obtención de Máscaras

Contexto Teórico y Simulaciones: Este algoritmo convierte la imagen original al espacio de color HSV para mejorar el contraste entre las fresas y el fondo, lo que facilita la segmentación mediante clusterización de colores. La simulación mostró que esta conversión es crucial para la detección eficaz bajo diversas condiciones de iluminación.

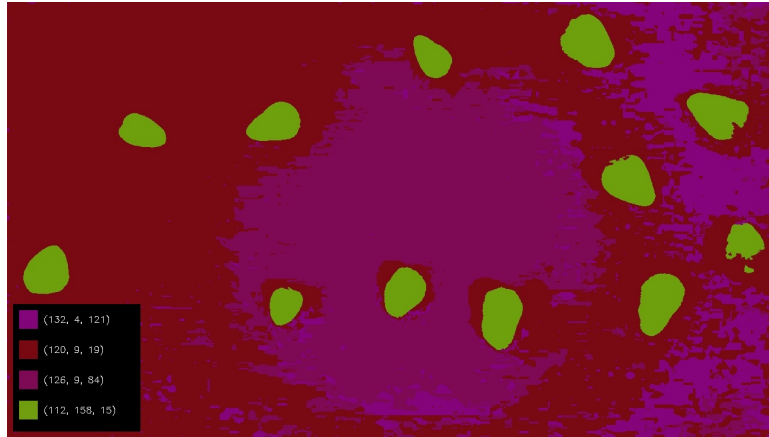


Figura 6.1: Imagen Original en Espacio de Color HSV y Clusterizada.

El primer producto obtenido del algoritmo es la imagen clusterizada (Fig. 6.1.), después de la lectura de la imagen se hace una transformación a espacio de color HSV sobre el cual se hace un posterior proceso de simplificación. La imagen en HSV no se guarda porque no es un resultado que ofrezca información relevante para el procedimiento, sin embargo, la imagen clusterizada y su información es esencial para la constitución de la máscara.

$$inferior = (9, 110, 103) \quad (6.1)$$

$$superior = (18, 194, 113) \quad (6.2)$$

El proceso de clusterización permite simplificar la imagen en algunos de sus colores representativos, para este caso, el algoritmo estaba instruido para identificar los cuatro tonos mayoritarios en cada fotograma y hacia los cuales demás píxeles se van a promediar al más cercano.

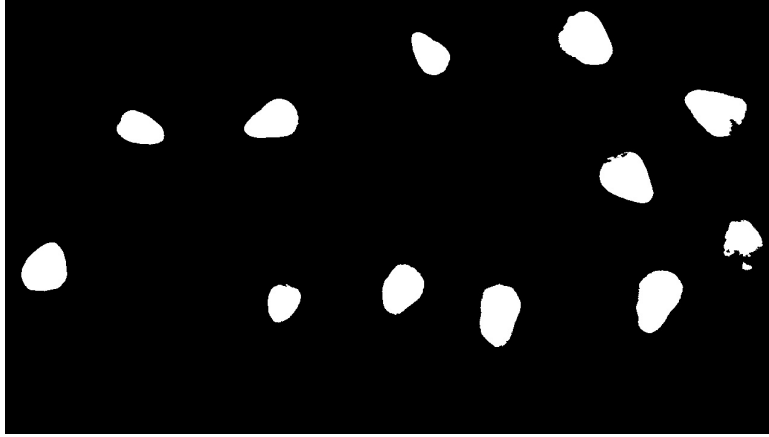


Figura 6.2: Imagen Original Convertida en Máscara.

Para poder definir el rango de valores que no se van a filtrar en la máscara, los que no se van a omitir, se registraron todos las tuplas que indicaban los tonos relacionados con las fresas (Tabla 6.1.). A partir de esos resultados, se determinan el límite superior e inferior del rango de colores que la máscara no va a omitir, para cada elemento en cada tupla se encuentran los valores menores y mayores que se registraron a través de la toma.

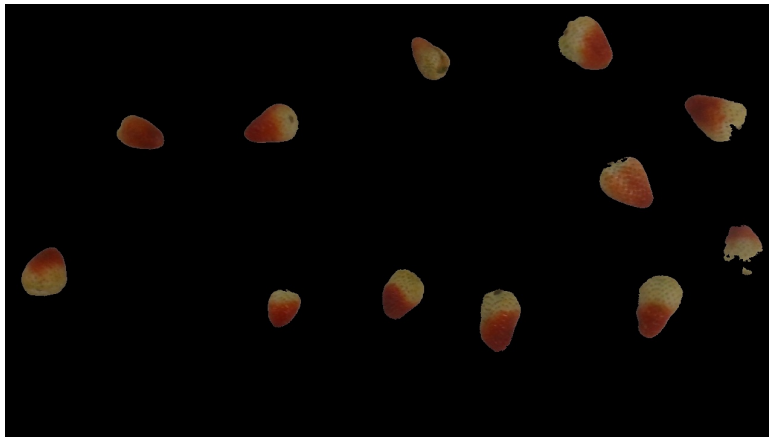


Figura 6.3: Máscara Aplicada a la Imagen Original.

Ya con esos rangos definidos, se le indica al algoritmo que debe excluir de la imagen todos los píxeles que no se encuentran dentro de esos valores y se crea una matriz binaria donde todos los píxeles que cumplen se transforman en unos (blancos) y los que no en ceros (negros).

Centroides Fresas	Fotograma	Centroides Fresas	Fotograma
(107, 165, 13)	ColorImage45.jpg	(107, 164, 13)	ColorImage46.jpg
(107, 165, 13)	ColorImage47.jpg	(107, 165, 12)	ColorImage48.jpg
(104, 194, 9)	ColorImage49.jpg	(108, 164, 12)	ColorImage50.jpg
(113, 111, 18)		-	-
(108, 164, 12)	ColorImage51.jpg	(109, 164, 12)	ColorImage52.jpg
(109, 163, 12)	ColorImage53.jpg	(108, 163, 12)	ColorImage54.jpg
(104, 192, 9)	ColorImage55.jpg	(109, 162, 12)	ColorImage56.jpg
(113, 110, 17)		-	-
(109, 163, 12)	ColorImage57.jpg	(109, 160, 12)	ColorImage58.jpg
(110, 159, 12)	ColorImage59.jpg	(110, 159, 12)	ColorImage60.jpg
(109, 160, 12)	ColorImage61.jpg	(108, 161, 12)	ColorImage62.jpg
(108, 158, 12)	ColorImage63.jpg	(108, 160, 12)	ColorImage64.jpg
(107, 164, 11)	ColorImage65.jpg	(107, 165, 11)	ColorImage66.jpg
(107, 165, 12)	ColorImage67.jpg	(106, 165, 12)	ColorImage68.jpg
(106, 165, 12)	ColorImage69.jpg	(104, 164, 12)	ColorImage70.jpg
(103, 167, 11)	ColorImage71.jpg	(103, 168, 11)	ColorImage72.jpg
(104, 168, 11)	ColorImage73.jpg	(104, 169, 11)	ColorImage74.jpg
(105, 170, 11)	ColorImage75.jpg	(104, 165, 11)	ColorImage76.jpg
(103, 167, 11)	ColorImage77.jpg	(103, 169, 10)	ColorImage78.jpg
(105, 172, 11)	ColorImage79.jpg	(106, 173, 11)	ColorImage80.jpg
(106, 171, 11)	ColorImage81.jpg	(106, 170, 11)	ColorImage82.jpg
(106, 170, 11)	ColorImage83.jpg	(109, 169, 11)	ColorImage84.jpg
(106, 169, 11)	ColorImage85.jpg	(106, 168, 11)	ColorImage86.jpg
(107, 166, 11)	ColorImage87.jpg	(107, 168, 11)	ColorImage88.jpg
(107, 164, 11)	ColorImage89.jpg	(108, 163, 12)	ColorImage90.jpg
(109, 162, 12)	ColorImage91.jpg	(109, 161, 11)	ColorImage92.jpg
(108, 161, 11)	ColorImage93.jpg	(109, 160, 11)	ColorImage94.jpg
(110, 159, 11)	ColorImage95.jpg	(110, 158, 11)	ColorImage96.jpg
(110, 158, 11)	ColorImage97.jpg	(110, 158, 11)	ColorImage98.jpg
(109, 156, 11)	ColorImage99.jpg	(109, 155, 11)	ColorImage100.jpg
(108, 153, 11)	ColorImage101.jpg	(106, 154, 11)	ColorImage102.jpg
(105, 160, 10)	ColorImage103.jpg	(105, 163, 9)	ColorImage104.jpg
(105, 163, 9)	ColorImage105.jpg	(106, 160, 9)	ColorImage106.jpg
(106, 157, 10)	ColorImage107.jpg	(107, 158, 10)	ColorImage108.jpg
(104, 156, 10)	ColorImage109.jpg	(107, 153, 10)	ColorImage110.jpg
(109, 150, 10)	ColorImage111.jpg	(110, 147, 10)	ColorImage112.jpg
(110, 147, 10)	ColorImage113.jpg	-	-

Cuadro 6.1: Centroides de los Clusteres de las Fresas

Así, cuando se aplica una operación entre la imagen original y la máscara, los valores de píxeles que se multiplican por el uno van a mantenerse en la imagen resultante, por cero, se omiten.

Finalmente, los otros dos productos obtenidos son la máscara (Fig. 6.2.) y la imagen original con la máscara aplicada (Fig. 6.3.). Las tres imágenes se almacenan en su respectiva carpeta con el nombre con el que se encuentra registrado el fotograma original, y en un directorio análogo al del proyecto fuente.

El procesamiento de 69 fotogramas tomó 67.281 segundos en producir 69 imágenes en espacio de color HSV, 69 máscaras y 69 copias del original con la máscara aplicada, un total de 207 archivos a 3.076 copias por segundo, todas del mismo tamaño.

$$69HSV + 69Máscaras + 69MáscarasAplicadas = 207Copias \quad (6.3)$$

$$ImágenesProcesadas = \frac{207Copias}{67,281s} = 3,076 \frac{Copias}{s} \quad (6.4)$$

6.2. Resultados del Algoritmo de Identificación, Seguimiento y Extracción de Objetos

El primer producto del segundo algoritmo es una copia de la máscara fuente cambiada a escala de grises, sin embargo, esta copia de la foto no se guarda porque su función en el código requiere dicha transformación que es equivalente a una matriz con la cantidad de elementos que contiene ese tipo de imágenes para que la librería detecte los contrornos.

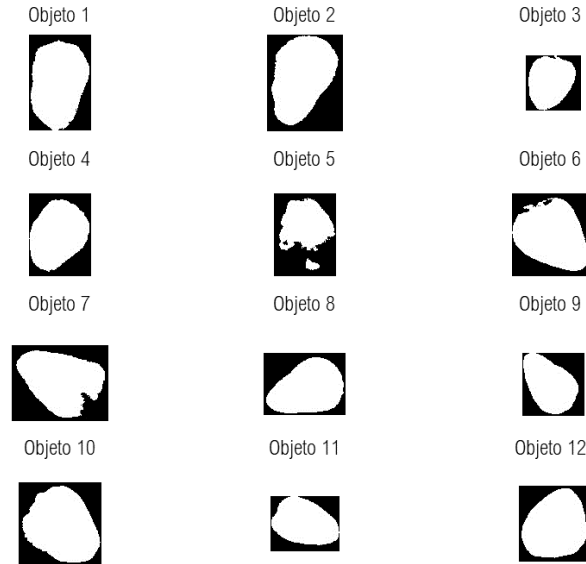


Figura 6.4: Objetos Detectados Extraídos en la Imagen Original.

El segundo producto sí es material relevante para el proceso de investigación, se genera en una carpeta para cada fotograma con un recorte de cada fruta identificada por el algoritmo (Fig. 6.4.) en dicha imagen. En la foto de referencia, identificada como ColorImage58.jpg, se identificaron doce objetos y se etiquetaron con respecto a los algoritmos de seguimiento.

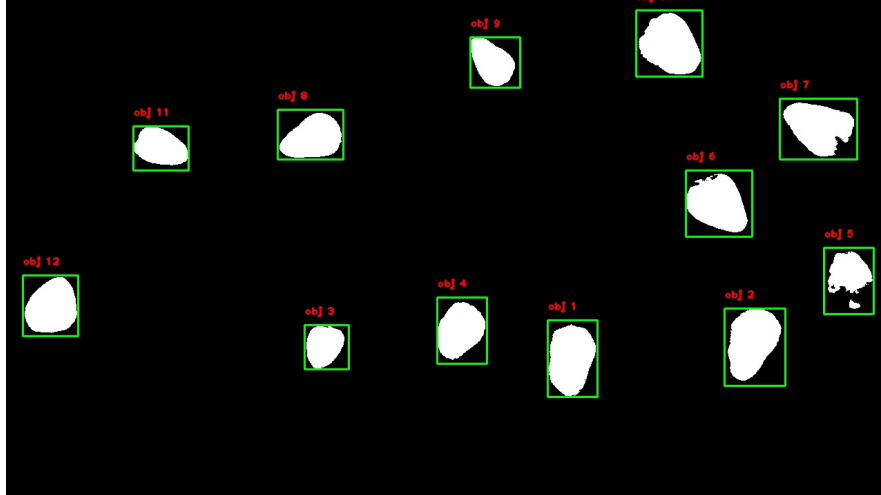


Figura 6.5: Imagen Original con los Objetos Detectados Resaltados.

El tercer producto es una copia de la máscara de cada fotograma donde se resalta la caja delimitadora y el identificador correspondiente a cada fruta detectada; esta copia permite realizar un proceso de verificación con respecto al producto anterior en la medida en la que permite corroborar los objetos en cantidad, forma e identificador.

El cuarto y último producto de la ejecución de este algoritmo es archivo CSV donde se almacena la información de toda la lectura en tablas de texto (Tabla 6.2.), más específicamente la información morfológica de cada fresa que indica en que fotograma fue registrada, cuál es su identificador, el área, caja delimitadora y centroide.

Los otros algoritmos relevantes para esta etapa no tienen productos propios resultado de su ejecución, el algoritmo de limpieza de datos, en el caso de ser necesario, borra las imágenes sin información, completamente negras, que pueden presentar un problema para la ejecución.

6.2.1. Centroid Tracker

Contexto Teórico y Simulaciones: Basado en la distancia euclidiana, este rastreador evalúa la proximidad de los centroides para mantener la continuidad del seguimiento de objetos a través de los fotogramas. Las simulaciones validan su eficacia en secuencias donde los objetos se mueven moderadamente y mantienen una proximidad constante.

El procesamiento de 69 fotogramas tomó 23.141 segundos en producir 525 recortes de frutas y 69 copias de las máscara con los identificadores y las cajas delimitadoras resaltadas, un total de 594 archivos a 25.668 copias por segundo. Aunque es necesario tener en cuenta que los 525 recortes son imágenes de bastante menor tamaño, causal del aparente aumento de eficiencia con

Frame	Object	Centroid	BoundingBox	Area
ColorImage45.jpg	1	(482, 559)	(448, 504, 520, 624)	5487
ColorImage45.jpg	2	(740, 545)	(704, 496, 784, 608)	5391
ColorImage45.jpg	3	(136, 526)	(104, 496, 176, 560)	3061.5
ColorImage45.jpg	4	(329, 512)	(288, 464, 376, 560)	4921
ColorImage45.jpg	5	(893, 450)	(856, 408, 928, 488)	3347
ColorImage45.jpg	6	(717, 338)	(672, 288, 760, 384)	5364.5
ColorImage45.jpg	7	(875, 230)	(832, 192, 928, 280)	4997
ColorImage45.jpg	8	(155, 216)	(104, 176, 200, 248)	4720.5
ColorImage45.jpg	9	(420, 114)	(392, 72, 456, 152)	3218.5
ColorImage45.jpg	10	(674, 100)	(624, 48, 720, 152)	5789
ColorImage46.jpg	1	(517, 555)	(480, 504, 552, 616)	5333.5
ColorImage46.jpg	2	(774, 539)	(736, 488, 816, 600)	5454
ColorImage46.jpg	3	(169, 520)	(136, 488, 208, 560)	3158.5
ColorImage46.jpg	4	(364, 506)	(328, 464, 408, 552)	4759.5
ColorImage46.jpg	5	(925, 448)	(888, 408, 960, 504)	3556
ColorImage46.jpg	6	(748, 334)	(704, 288, 792, 384)	5346.5
ColorImage46.jpg	7	(906, 226)	(856, 184, 960, 272)	5007
ColorImage46.jpg	8	(184, 214)	(136, 176, 232, 248)	4581
ColorImage46.jpg	9	(450, 112)	(416, 72, 488, 152)	3298
ColorImage46.jpg	10	(704, 97)	(656, 48, 752, 144)	5798
ColorImage47.jpg	1	(554, 544)	(520, 488, 592, 605)	5184.5
ColorImage47.jpg	2	(811, 527)	(776, 480, 856, 584)	5329.5
ColorImage47.jpg	3	(205, 515)	(176, 480, 240, 552)	2909
ColorImage47.jpg	4	(401, 499)	(360, 456, 448, 544)	4600
ColorImage47.jpg	5	(961, 436)	(928, 392, 1000, 488)	3698
ColorImage47.jpg	6	(782, 324)	(736, 280, 832, 368)	5374.5
ColorImage47.jpg	11	(19, 223)	(0, 192, 48, 251)	1944.5
ColorImage47.jpg	7	(938, 216)	(891, 176, 992, 264)	5299
ColorImage47.jpg	8	(216, 209)	(168, 168, 264, 240)	4615
ColorImage47.jpg	9	(482, 107)	(450, 64, 520, 144)	3132
ColorImage47.jpg	10	(735, 90)	(688, 40, 784, 136)	5775
ColorImage48.jpg	1	(575, 536)	(544, 480, 616, 592)	5183.5
ColorImage48.jpg	3	(225, 511)	(192, 480, 264, 552)	2995.5
ColorImage48.jpg	2	(831, 517)	(792, 464, 872, 576)	5356.5
ColorImage48.jpg	4	(421, 492)	(384, 448, 464, 536)	4647
ColorImage48.jpg	5	(980, 425)	(944, 384, 1016, 477)	3794.5
ColorImage48.jpg	6	(800, 316)	(752, 272, 848, 360)	5291
ColorImage48.jpg	11	(27, 219)	(0, 184, 64, 248)	2868
ColorImage48.jpg	7	(956, 206)	(904, 168, 1008, 256)	5351
ColorImage48.jpg	8	(234, 206)	(184, 168, 280, 240)	4558.5
ColorImage48.jpg	9	(499, 102)	(464, 64, 536, 144)	3204.5
ColorImage48.jpg	10	(751, 83)	(704, 32, 800, 128)	5785
...	...	...	...	...

Cuadro 6.2: Muestra de los Datos Registrados

respecto a la obtención de máscaras.

$$525Recortes + 69ImágenesResaltadas = 594Copias \quad (6.5)$$

$$ImágenesProcesadas = \frac{594copias}{23,141s} = 25,668 \frac{Copias}{s} \quad (6.6)$$

6.2.2. Bounding Box Tracker

Contexto Teórico y Simulaciones: Este método utiliza las esquinas de las cajas delimitadoras para rastrear objetos, proporcionando una perspectiva más geométrica que el CentroidTracker. El tiempo de simulación registrado indica una peor respuesta en entornos donde los objetos cambian de orientación o escala, aunque nada conclusivo.

El procesamiento de 69 fotogramas tomó 23.712 segundos en producir 525 recortes de frutas y 69 copias de las máscara con los identificadores y las cajas delimitadoras resaltadas, un total de 594 archivos a 25.05 copias por segundo. Aunque es necesario tener en cuenta que los 525 recortes son imágenes de bastante menor tamaño, causal del aparente aumento de eficiencia con respecto a la obtención de máscaras.

$$525Recortes + 69ImágenesResaltadas = 594Copias \quad (6.7)$$

$$ImágenesProcesadas = \frac{594copias}{23,712s} = 25,05 \frac{Copias}{s} \quad (6.8)$$

6.2.3. Intersection Over Union Tracker

Contexto Teórico y Simulaciones: Este algoritmo utiliza el cálculo de intersección sobre unión para identificar continuidad en el seguimiento. Este método mostró ser el más rápido en las simulaciones, especialmente efectivo en entornos dinámicos donde los objetos experimentan cambios significativos en la posición o la forma entre los fotogramas, pero no de posición.

El procesamiento de 69 fotogramas tomó 21.882 segundos en producir 525 recortes de frutas y 69 copias de las máscara con los identificadores y las cajas delimitadoras resaltadas, un total de 594 archivos a 27.145 copias por segundo. Aunque es necesario tener en cuenta que los 525 recortes son imágenes de bastante menor tamaño, causal del aparente aumento de eficiencia.

$$525Recortes + 69ImágenesResaltadas = 594Copias \quad (6.9)$$

$$ImágenesProcesadas = \frac{594copias}{21,882s} = 27,145 \frac{Copias}{s} \quad (6.10)$$

6.3. Resultados Algoritmo de Cálculo del estado de Madurez

Contexto Teórico y Simulaciones: Este algoritmo aplica k-means clustering para segmentar las áreas de color dentro de una imagen, identificando zonas que indican diferentes estados de madurez en fresas. Las simulaciones demostraron que el análisis preciso de los clústeres de color es fundamental para estimar correctamente la madurez, basándose en la proporción de colores

característicos de las fresas maduras e inmaduras.

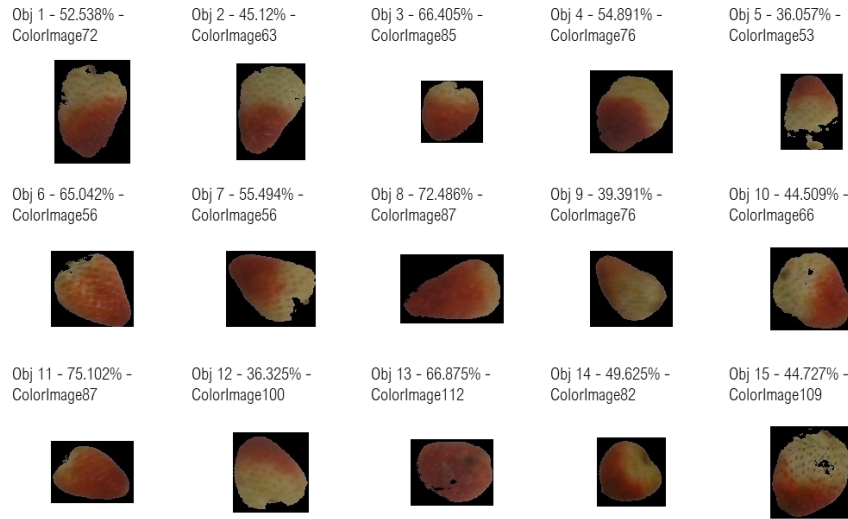


Figura 6.6: Objetos Detectados en la Toma Completa.

El algoritmo identifica la copia representativa de la fruta seleccionando en el archivo CSV una lista de las filas para las frutas del mismo identificador, compara las áreas y escoge la de mayor tamaño. Así, para cada identificador se guarda el recorte correspondiente a la fruta en la imagen cuyo nombre de archivo está en el respectivo registro (Fig. 6.6.).

Cada imagen se guarda además con un formato de nombre de archivo en el cual el primer argumento es el identificador del objeto, el segundo argumento es la estimación de la madurez para la fruta en cuestión y de último, el nombre del archivo del cual se obtuvo el recorte; ese último dato con el objetivo de brindarle al usuario la oportunidad de revisar el conjunto de datos comparando el recorte directamente con la foto donde fue extraído.

Por último se genera otro archivo CSV con la misma información del archivo CSV resultante del segundo algoritmo. El archivo contiene también los datos morfológicos de los objetos resultantes de toda la toma, centroide, área y caja delimitadora, el identificador del objeto que corresponde a su vez a su conteo en toda la toma, y al fotograma donde se encuentra (Tabla 6.3.).

Frame	Object	Centroid	BoundingBox	Area	Ripening
ColorImage72.jpg	1	(1225, 561)	(1184, 504, 1272, 624)	6512.5	0.525379468
ColorImage55.jpg	2	(1003, 499)	(967, 448, 1048, 560)	6127	0.411079687
ColorImage85.jpg	3	(1199, 586)	(1168, 552, 1240, 624)	3678	0.664048866
ColorImage76.jpg	4	(1198, 536)	(1152, 488, 1248, 584)	5632.5	0.548913043
ColorImage49.jpg	5	(997, 425)	(960, 376, 1040, 472)	5179.5	0.383220911
ColorImage55.jpg	6	(956, 300)	(904, 256, 1008, 344)	5873.5	0.657062047
ColorImage55.jpg	7	(1107, 192)	(1056, 152, 1160, 240)	6010.5	0.529653051
ColorImage87.jpg	8	(1209, 228)	(1144, 184, 1264, 264)	5909	0.724855282
ColorImage76.jpg	9	(1213, 115)	(1168, 72, 1264, 160)	4224.5	0.393909384
ColorImage55.jpg	10	(893, 73)	(842, 24, 944, 120)	6218	0.471604118
ColorImage87.jpg	11	(942, 231)	(896, 192, 992, 264)	4165.5	0.75101833
ColorImage100.jpg	12	(1215, 440)	(1176, 392, 1264, 485)	5520	0.363252781
ColorImage112.jpg	13	(1194, 363)	(1144, 328, 1240, 405)	5027.5	0.668751303
ColorImage82.jpg	14	(252, 137)	(216, 96, 296, 176)	4207	0.496248383
ColorImage109.jpg	15	(854, 571)	(808, 517, 904, 624)	7136	0.447269445

Cuadro 6.3: Datos de Madurez

Capítulo 7

Discusión

7.1. Clasificación de las Técnicas de Detección y Seguimiento

Análisis de Técnicas: Las técnicas de seguimiento utilizadas en este estudio, CentroidTracker, BoundingBoxTracker, e IntersectionTracker, muestran diferencias significativas en su aplicabilidad según el tipo de movimiento y la naturaleza de los objetos en las imágenes. El CentroidTracker es eficiente para objetos que mantienen una trayectoria constante y predecible, mientras que el BoundingBoxTracker es más robusto frente a cambios de orientación o escala. El IntersectionTracker se distingue por su habilidad para manejar situaciones donde los objetos cambian drásticamente entre fotogramas, como en entornos dinámicos.

Impacto en Resultados: La elección de la técnica de seguimiento influye directamente en la precisión del seguimiento y, por tanto, en la calidad de los datos recopilados para análisis posteriores, como el cálculo de madurez. La eficacia de cada técnica depende críticamente del contexto de la aplicación, y este estudio ayuda a identificar qué técnica se adapta mejor a cada situación específica.

7.2. Acerca del Cálculo de Madurez

Metodología y Validación: El cálculo de madurez se basa en la segmentación de colores a través de k-means clustering, lo que permite una evaluación detallada del estado de madurez de las fresas. Esta técnica asume que la madurez puede inferirse de la proporción de colores en la imagen, lo cual es una aproximación general que puede no ser válida para todos los tipos de frutas o condiciones de iluminación.

Implicaciones Prácticas: Si bien el algoritmo demostró ser eficaz en las condiciones experimentales definidas, su aplicabilidad en entornos naturales o en diferentes tipos de cultivos puede requerir ajustes significativos. Esto subraya la necesidad de calibración cuidadosa y validación en diversos entornos para asegurar su precisión y utilidad práctica.

7.3. Límites Experimentales

Restricciones de Datos: Los resultados obtenidos están limitados por la calidad y la variedad de los datos de imagen utilizados en las simulaciones. Por ejemplo, la uniformidad en las condiciones de iluminación y la calidad de las imágenes pueden afectar significativamente la precisión del seguimiento y la clasificación de la madurez.

Generalización de Modelos: Los modelos desarrollados se basan en un conjunto limitado de datos y condiciones experimentales. La generalización de estos modelos a otras situaciones puede no ser directa y podría requerir adaptaciones o mejoras para manejar variaciones en tipos de objetos, condiciones ambientales o diferencias en el comportamiento de los objetos.

Retos Técnicos: La implementación de estos algoritmos en un entorno de producción real también presenta desafíos técnicos, como la necesidad de procesamiento en tiempo real, manejo de grandes volúmenes de datos y la integración con otros sistemas de manejo y procesamiento de datos.

Capítulo 8

Conclusiones

8.1. Eficiencia de los Algoritmos de Seguimiento

Los resultados demuestran que cada algoritmo de seguimiento ofrece ventajas distintas dependiendo de la dinámica y características de los objetos a rastrear. El IntersectionTracker proporcionó el tiempo de procesamiento más rápido y mostró superioridad en entornos donde los objetos presentan cambios significativos de posición o forma. Esta distinción subraya la importancia de seleccionar la técnica de seguimiento más adecuada basada en las especificidades del escenario de aplicación.

8.2. Viabilidad del Cálculo de Madurez Basado en Color

El uso de la clusterización por k-means para determinar la madurez de las fresas demostró ser efectivo bajo condiciones controladas. Sin embargo, la dependencia de la proporción de colores como indicador de madurez implica que cambios en la iluminación o variaciones en las características visuales de las fresas pueden afectar la precisión de los resultados. Esta técnica, aunque prometedora, requiere más validación y posiblemente ajustes para garantizar su aplicabilidad en entornos naturales y con diferentes tipos de cultivos.

8.3. Retos y Limitaciones Experimentales

Las simulaciones y pruebas realizadas revelan que las condiciones experimentales, como la calidad de imagen y la iluminación, juegan un papel crucial en la eficacia de los algoritmos. Los modelos y técnicas descritos son potencialmente limitados por la especificidad de los datos y las condiciones bajo las cuales fueron validados. Para una implementación real, se necesitaría una fase extensiva de pruebas en diferentes condiciones para adaptar y calibrar los algoritmos adecuadamente.

Bibliografía

- [1] Computer vision and image processing: A beginner's guide. <https://www.opencv.org>. Accessed: 2024-05-03.
- [2] Image recognition: Definition, algorithms & uses. <https://www.v7labs.com>. Accessed: 2024-05-03.
- [3] What is computer vision? <https://www.ibm.com>. Accessed: 2024-05-03.
- [4] What is computer vision? (definition, examples, uses). <https://builtin.com>. Accessed: 2024-05-03.
- [5] What is image recognition? <https://builtin.com>. Accessed: 2024-05-03.
- [6] The Algorithms. Rgb hsv conversion. <https://abc-algorithms.vercel.app/rgb-hsv-conversion>. Accessed: 2024-05-03.
- [7] Anonymous. Simple open-vocabulary object detection with vision transformers. *arXiv*, 2205.06230, 2022. Accessed: 2024-05-03.
- [8] Anonymous. Vision-based cranberry crop ripening assessment, 2023. arXiv:2309.00028. Disponible en <https://arxiv.labs.arxiv.org/html/2309.00028>.
- [9] R. Azadnia, S. Fouladi, and A. Jahanbakhshi. Intelligent detection and waste control of hawthorn fruit based on ripening level using machine vision system and deep learning techniques. *Results in Engineering*, 17:100891, 2023. Disponible en <https://www.journals.elsevier.com/results-in-engineering>.
- [10] CloudFactory. Iou (intersection over union). <https://wiki.cloudfactory.com/docs/mp-wiki/metrics/iou-intersection-over-union>. Accessed: 2024-05-03.
- [11] F.G. Costa, R.M. Silva, and L.S. Oliveira. Aplicación de redes neuronales convolucionales para la determinación de la madurez de bananas. *Revista Brasileira de Automatización Agrícola*, 3(2):34–42, 2022.
- [12] Mathematics Stack Exchange. Rgb to hsv color conversion algorithm. <https://math.stackexchange.com/questions/rgb-to-hsv-color-conversion-algorithm>. Accessed: 2024-05-03.
- [13] James D. Foley, Andries van Dam, Steven K. Feiner, and John F. Hughes. *Computer Graphics: Principles and Practice*. Addison-Wesley, 2nd edition, 1996.
- [14] Juan García and María Martínez. Traditional manual method for strawberry ripening classification. *Agricultural Sciences Review*, 15(4):321–328, 2021.

- [15] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*. Prentice Hall, 2nd edition, 2002.
- [16] María González and Javier López. A review of the hsv color model: Properties, applications, and challenges. *IEEE Transactions on Image Processing*, 24(6):1820–1835, 2015.
- [17] María González and Javier López. Automatización del conteo de fresas en invernaderos utilizando visión por computadora. *Revista Española de Agricultura*, 45(2):87–95, 2023.
- [18] Phil Green. *Digital Color Management: Encoding Solutions*. John Wiley & Sons, 2007.
- [19] Juan Hernández and Ana García. Optimización del riego en cultivos de fresas mediante análisis de imágenes. *Revista Mexicana de Agricultura*, 17(3):45–52, 2022.
- [20] I. T. Jolliffe and J. Cadima. Principal component analysis: A review and recent developments. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 374(2065):20150202, 2016.
- [21] LearnOpenCV. The complete guide to object tracking – opencv, deepsort, fairmot. <https://www.learnopencv.com/object-tracking-guide-opencv-deepsort-fairmot>. Accessed: 2024-05-03.
- [22] LearnOpenCV. Moving object detection using opencv. <https://www.learnopencv.com/moving-object-detection-using-opencv>. Accessed: 2024-05-03.
- [23] Y. Li, H. Zhang, and X. Shen. A survey of multi-view machine learning. *Neural Computing and Applications*, 23(7-8):2031–2038, 2013.
- [24] J. B. MacQueen. Some methods for classification and analysis of multivariate observations. 1:281–297, 1967.
- [25] J. Martínez-Pérez, F. García-Ruiz, and L. Gómez-Robledo. Implementación de técnicas de visión por computadora para la evaluación de calidad en uvas de vinificación. *Tecnología y Ciencias del Vino*, 18(1):18–26, 2023.
- [26] matplotlib. matplotlib (Versión 4.5) [Software], 2023. Disponible en <https://matplotlib.org/stable/index.html>.
- [27] mechaphantom. Simple object detection opencv. <https://github.com/mechaphantom/Simple-Object-Detection-OpenCV>. Accessed: 2024-05-03.
- [28] Jan Morovic and Wolfgang Sachtler. Understanding color management. *Journal of Electronic Imaging*, 18(3):031211, 2009.
- [29] NumPy. NumPy (Versión 1.24) [Software], 2023. Disponible en <https://numpy.org/doc/1.24>.
- [30] OpenCV. OpenCV: Open Source Computer Vision Library (Versión 4.7) [Software], 2023. Disponible en <https://pypi.org/project/opencv-python/>.
- [31] A. B. Payne, K. B. Walsh, P. P. Subedi, and D. Jarvis. Estimation of mango crop yield using image analysis – segmentation method. *Computers and Electronics in Agriculture*, 91:57–64, 2013.

- [32] PyImageSearch. Intersection over union (iou) for object detection. <https://pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/>, November 2016. Accessed: 2024-05-03.
- [33] PyImageSearch. Simple object tracking with opencv. <https://pyimagesearch.com/2018/07/23/simple-object-tracking-with-opencv/>, July 2018. Accessed: 2024-05-03.
- [34] Python Software Foundation. Python (Versión 3.11) [Software], 2023. Disponible en <https://www.python.org/doc>.
- [35] B. Santhi, R. Manikandan, M. Rahimi, and A. H. Gandomi. Computer vision system for mango fruit defect detection using deep convolutional neural network. *Foods*, 11(21):3483, 2022. Disponible en <https://doi.org/10.3390/foods11213483>.
- [36] Emily Smith and David Johnson. Color analysis for strawberry ripening classification. *Journal of Agricultural Science*, 27(3):215–222, 2019.
- [37] Emily Smith and David Johnson. Detección temprana de enfermedades en fresas mediante imágenes aéreas. *Journal of Agricultural Technology*, 39(1):78–86, 2023.
- [38] Viso Suite. Image recognition in 2024: A comprehensive guide. 2024. Accessed: 2024-05-03.
- [39] Richard Szeliski. *Computer Vision: Algorithms and Applications*. Springer, 2010.
- [40] Yoko Tanaka and Koji Suzuki. Evaluación automática de la calidad de fresas mediante imágenes multiespectrales y aprendizaje profundo. *Journal of Agricultural Informatics*, 12:45–59, 2021.
- [41] A.J. Thompson and M. H. Lee. Desarrollo de un sistema de clasificación automatizado para cerezas basado en visión por computadora. *Journal of Food Engineering*, 287:110115, 2021.
- [42] Xin Wang and Wei Zhang. Applications of the hsv color space in image processing. *Journal of Visual Communication and Image Representation*, 52:17–31, 2018.
- [43] Wikipedia. Clipping path. https://en.wikipedia.org/wiki/Clipping_path. Accessed: 2024-05-03.
- [44] Wikipedia. Cluster analysis. https://en.wikipedia.org/wiki/Cluster_analysis. Accessed: 2024-05-03.
- [45] Wikipedia. Color model. https://en.wikipedia.org/wiki/Color_model. Accessed: 2024-05-03.
- [46] Wikipedia. Color space. https://en.wikipedia.org/wiki/Color_space. Accessed: 2024-05-03.
- [47] Wikipedia. Computer vision. https://en.wikipedia.org/wiki/Computer_vision. Accessed: 2024-05-03.
- [48] Wikipedia. Hsl and hsv. https://en.wikipedia.org/wiki/HSL_and_HSV. Accessed: 2024-05-03.
- [49] Wikipedia. Hsl color solid cylinder saturation gray. https://en.wikipedia.org/wiki/HSL_and_HSV#/media/File:HSL_color_solid_cylinder_saturation_gray.png. Accessed: 2024-05-03.

- [50] Wikipedia. Image segmentation. https://en.wikipedia.org/wiki/Image_segmentation. Accessed: 2024-05-03.
- [51] Wikipedia. Jaccard index. https://en.wikipedia.org/wiki/Jaccard_index. Accessed: 2024-05-03.
- [52] Wikipedia. Mask (computing). [https://en.wikipedia.org/wiki/Mask_\(computing\)](https://en.wikipedia.org/wiki/Mask_(computing)). Accessed: 2024-05-03.
- [53] Wikipedia. Moving object detection. https://en.wikipedia.org/wiki/Moving_object_detection. Accessed: 2024-05-03.
- [54] Wikipedia. Object detection. https://en.wikipedia.org/wiki/Object_detection. Accessed: 2024-05-03.
- [55] Wikipedia. Open-vocabulary object detection. https://en.wikipedia.org/wiki/Open-vocabulary_object_detection. Accessed: 2023-05-03.