



## Proyecto Final de Estudios

Cosechador de Lechugas Autónomo con Unidad de  
Detección por Inteligencia Artificial

Brenda Gudiño  
Alan Vignolo

Fecha de presentación  
XX/XX/2025

# Índice general

# Resumen

# **1. Introducción**

## **1.1. Contexto y Motivación**

## **1.2. Objetivos del Proyecto**

## **1.3. Alcance y Limitaciones**

## **1.4. Estructura del Documento**

## 2. Marco Teórico

### 2.1. Sistemas de Control Jerárquico

### 2.2. Fundamentos de Visión Artificial y Aprendizaje Profundo

#### 2.2.1. Procesamiento Digital de Imágenes

El procesamiento digital de imágenes constituye el fundamento tecnológico de los sistemas de visión artificial en robótica agrícola. Esta disciplina integra conceptos de análisis matemático, álgebra lineal y teoría de señales para transformar datos visuales crudos en información estructurada y procesable por algoritmos de decisión.

#### Representación Digital de Imágenes

Una imagen digital se define matemáticamente como una función bidimensional discreta  $f(x, y)$ , donde las coordenadas espaciales  $(x, y)$  representan posiciones en el plano de la imagen y el valor de la función indica la intensidad luminosa en cada punto. En el contexto de imágenes a color en formato RGB, esta representación se extiende a un espacio tridimensional:

$$\mathbf{I}(x, y) = \begin{bmatrix} R(x, y) \\ G(x, y) \\ B(x, y) \end{bmatrix} \in [0, 255]^3 \quad (2.1)$$

donde  $R$ ,  $G$  y  $B$  representan los canales de color rojo, verde y azul respectivamente. Esta representación permite codificar aproximadamente 16.7 millones de colores distintos ( $256^3$ ), cubriendo ampliamente el espectro visible necesario para aplicaciones de agricultura de precisión.

La resolución espacial de una imagen determina la cantidad de información disponible para el análisis. Una imagen de dimensiones  $M \times N$  píxeles contiene  $M \cdot N$  elementos de información, cada uno representando el promedio de la radiancia incidente sobre el área del sensor correspondiente a ese píxel.

#### Espacios de Color y Transformaciones

El espacio de color RGB, aunque intuitivo y ampliamente utilizado en sistemas de captura, presenta limitaciones para ciertas tareas de procesamiento debido a su fuerte correlación entre canales y sensibilidad a variaciones de iluminación. Por esta razón, la conversión a espacios de color alternativos resulta fundamental en visión por computadora.

##### Espacio HSV (Hue, Saturation, Value)

El espacio HSV separa la información cromática (matiz y saturación) de la información de iluminación (valor), proporcionando robustez frente a cambios en las condiciones de luz. La transformación de RGB a HSV se define mediante:

$$V = \max(R, G, B) \quad (2.2)$$

$$S = \begin{cases} \frac{V - \min(R, G, B)}{V} & \text{si } V \neq 0 \\ 0 & \text{si } V = 0 \end{cases} \quad (2.3)$$



Figura 2.1: Representación de una imagen digital en el espacio RGB mostrando los tres canales de color

$$H = \begin{cases} 60 \cdot \frac{G-B}{V-\min(R,G,B)} & \text{si } V = R \\ 60 \cdot \left( 2 + \frac{B-R}{V-\min(R,G,B)} \right) & \text{si } V = G \\ 60 \cdot \left( 4 + \frac{R-G}{V-\min(R,G,B)} \right) & \text{si } V = B \end{cases} \quad (2.4)$$

Esta descomposición resulta particularmente ventajosa para la detección de elementos basada en contraste, ya que el canal  $V$  captura la intensidad luminosa independientemente del color, permitiendo operaciones de umbralización más robustas.

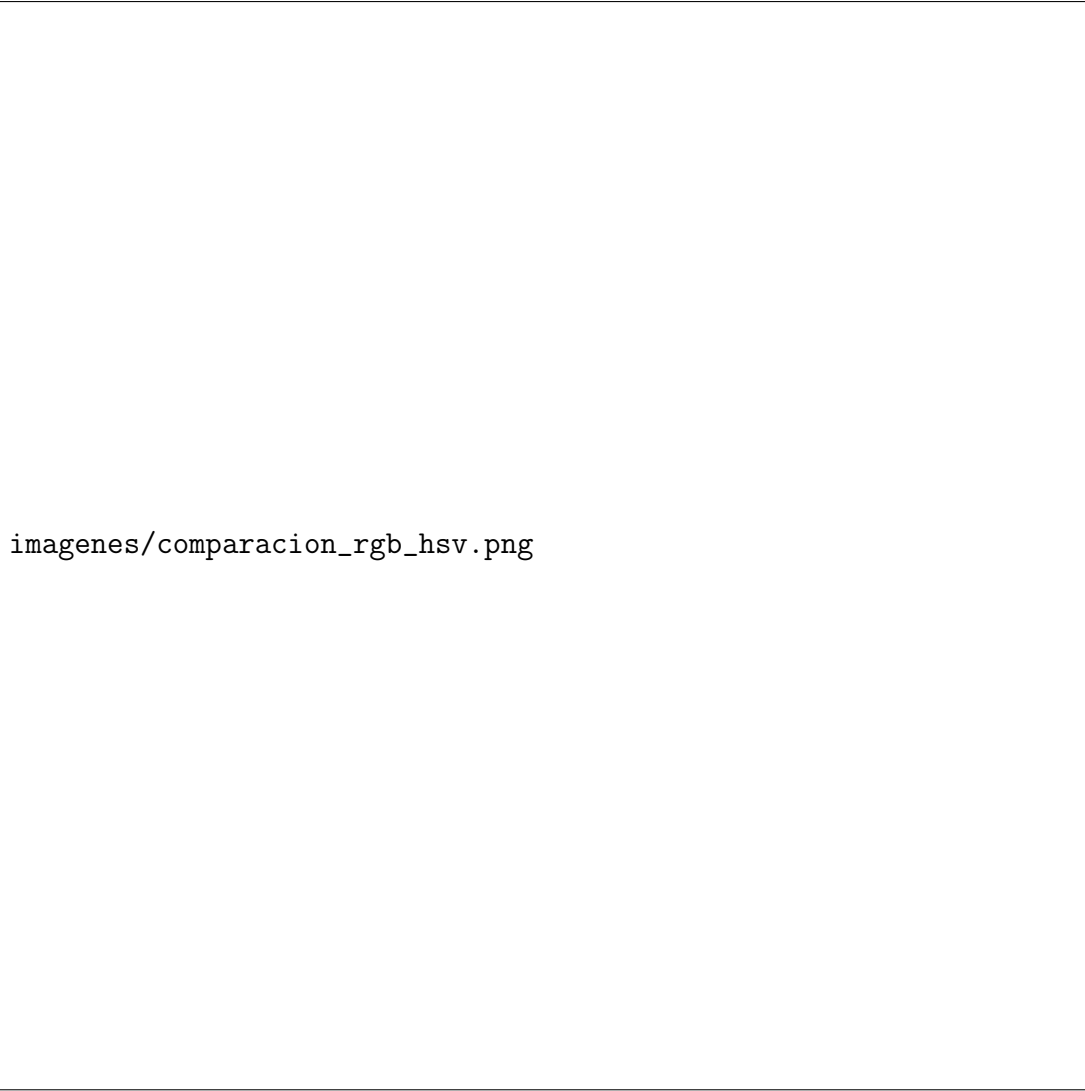
### Ventajas Operativas del Espacio HSV

La separación de la información de brillo en el canal  $V$  permite implementar algoritmos de detección invariantes a cambios de iluminación ambiental. En aplicaciones agrícolas donde las condiciones lumínicas varían significativamente durante el día, esta propiedad resulta crítica para mantener la consistencia operativa del sistema.

Matemáticamente, el canal  $V$  satisface la propiedad de invariancia:

$$V(\alpha \cdot \mathbf{I}_{RGB}) = \alpha \cdot V(\mathbf{I}_{RGB}) \quad \forall \alpha > 0 \quad (2.5)$$

donde  $\alpha$  representa un factor de escala de iluminación global. Esta linealidad permite compensar variaciones uniformes de brillo mediante ajustes simples de umbralización.



imagenes/comparacion\_rgb\_hsv.png

Figura 2.2: Comparación entre espacios de color RGB y HSV para una misma escena de cultivo

## Umbralización y Segmentación

La umbralización constituye una técnica fundamental de segmentación que particiona una imagen en regiones de interés mediante la comparación de intensidades de píxeles contra valores de referencia. La umbralización binaria se define formalmente como:

$$g(x, y) = \begin{cases} 255 & \text{si } f(x, y) > T \\ 0 & \text{si } f(x, y) \leq T \end{cases} \quad (2.6)$$

donde  $T$  es el valor umbral y  $g(x, y)$  es la imagen resultante.

### Umbralización Inversa

En escenarios donde los elementos de interés presentan intensidades menores que el fondo, la umbralización inversa resulta más apropiada:

$$g(x, y) = \begin{cases} 255 & \text{si } f(x, y) < T \\ 0 & \text{si } f(x, y) \geq T \end{cases} \quad (2.7)$$

Esta variante permite destacar regiones oscuras contra fondos claros, técnica ampliamente aplicada en la detección de marcadores de contraste en ambientes controlados.

### Selección Óptima del Umbral



Figura 2.3: Proceso de umbralización binaria inversa para destacar elementos oscuros

La determinación del valor umbral óptimo  $T^*$  puede realizarse mediante análisis del histograma de intensidades. El método de Otsu minimiza la varianza intra-clase de los grupos resultantes:

$$T^* = \arg \min_T [\omega_0(T)\sigma_0^2(T) + \omega_1(T)\sigma_1^2(T)] \quad (2.8)$$

donde  $\omega_i$  y  $\sigma_i^2$  representan el peso y varianza de cada clase. Alternativamente, puede emplearse conocimiento previo del entorno para establecer umbrales fijos que aprovechen características específicas del escenario operativo.

## Operaciones Morfológicas

Las operaciones morfológicas manipulan la forma y estructura de objetos en imágenes binarias mediante la aplicación de elementos estructurantes. Estas operaciones se fundamentan en la teoría de conjuntos y permiten eliminar ruido, rellenar huecos y separar objetos conectados.

### Erosión

La erosión reduce el tamaño de los objetos eliminando píxeles del perímetro:

$$(A \ominus B)(x, y) = \min_{(s,t) \in B} \{A(x + s, y + t)\} \quad (2.9)$$

donde  $A$  es la imagen binaria y  $B$  es el elemento estructurante.

### Dilatación

La dilatación expande los objetos añadiendo píxeles al perímetro:



$$(A \oplus B)(x, y) = \max_{(s,t) \in B} \{A(x + s, y + t)\} \quad (2.10)$$

### Apertura y Cierre

La apertura ( $A \circ B = (A \ominus B) \oplus B$ ) elimina pequeños objetos y protuberancias, mientras el cierre ( $A \bullet B = (A \oplus B) \ominus B$ ) rellena huecos pequeños y conecta componentes próximos. Estas operaciones compuestas resultan esenciales para el refinamiento de máscaras de segmentación en aplicaciones reales.



Figura 2.4: Efecto de operaciones morfológicas sobre una imagen binaria

## Calibración y Transformación de Coordenadas

La calibración establece la correspondencia entre coordenadas píxel en la imagen y coordenadas físicas en el espacio de trabajo. Esta transformación es fundamental para convertir mediciones visuales en comandos de movimiento precisos.

### Modelo de Transformación Lineal

Para configuraciones donde la cámara observa perpendicular al plano de trabajo, la transformación píxel-a-métrica puede aproximarse mediante un modelo lineal:

$$\begin{bmatrix} x_{mm} \\ y_{mm} \end{bmatrix} = \begin{bmatrix} k_x & 0 \\ 0 & k_y \end{bmatrix} \begin{bmatrix} x_{px} \\ y_{px} \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \quad (2.11)$$

## Proyecto Final de Estudios: Robot cosechador automático

donde  $k_x, k_y$  son factores de escala (mm/píxel) y  $t_x, t_y$  son desplazamientos de origen.

### Estimación por Mínimos Cuadrados

Los parámetros de calibración se estiman mediante regresión lineal a partir de puntos de correspondencia conocidos:

$$\mathbf{K} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y} \quad (2.12)$$

donde  $\mathbf{X}$  contiene las coordenadas píxel de puntos de referencia,  $\mathbf{y}$  sus coordenadas métricas medidas físicamente, y  $\mathbf{K}$  el vector de parámetros de calibración. Este método minimiza el error cuadrático medio garantizando la solución óptima en el sentido de mínimos cuadrados.



Figura 2.5: Proceso de calibración mediante puntos de correspondencia conocidos

### Validación Estadística

La precisión de la calibración se cuantifica mediante el error cuadrático medio (RMSE):

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2} \quad (2.13)$$

donde  $\hat{y}_i$  son las predicciones del modelo. Valores de RMSE inferiores a 1mm resultan aceptables para aplicaciones de manipulación agrícola de precisión.

### 2.2.2. Detección y Análisis de Contornos

La detección de contornos constituye una técnica fundamental en visión por computadora para la identificación y caracterización de objetos en imágenes. Los contornos representan las fronteras entre

## Proyecto Final de Estudios: Robot cosechador automático

regiones con propiedades visuales distintas, proporcionando información estructural esencial para el reconocimiento y clasificación de elementos.

### Definición Matemática de Contorno

Un contorno se define como la secuencia ordenada de píxeles que forman el límite de una región conexa en una imagen binaria:

$$C = \{(x_i, y_i)\}_{i=1}^n \quad \text{donde } (x_i, y_i) \in \partial R \quad (2.14)$$

siendo  $\partial R$  la frontera topológica de la región  $R$ . Formalmente, un punto  $(x, y)$  pertenece a  $\partial R$  si:

$$(x, y) \in R \quad \wedge \quad \exists (x', y') \in \mathcal{N}(x, y) : (x', y') \notin R \quad (2.15)$$

donde  $\mathcal{N}(x, y)$  representa el vecindario del píxel (usualmente 4-conectividad u 8-conectividad).

### Algoritmos de Detección de Contornos

#### Método de Seguimiento de Bordes (Border Following)

El algoritmo de Moore-Neighbor tracing recorre el perímetro de objetos binarios siguiendo el borde en sentido horario o antihorario:

1. Localizar primer píxel de objeto (barrido izquierda-derecha, arriba-abajo)
2. Examinar vecinos en orden circular desde el píxel de entrada
3. Seleccionar primer vecino que pertenezca al objeto
4. Repetir hasta retornar al píxel inicial

La complejidad computacional es  $\mathcal{O}(n)$  donde  $n$  es el número de píxeles del perímetro.

#### Algoritmo de Suzuki-Abe

Este método detecta simultáneamente contornos externos e internos (huecos), organizándolos jerárquicamente. Define dos tipos de bordes:

- **Border tipo 0:** Borde externo de objeto rodeado por fondo
- **Border tipo 1:** Borde interno de hueco rodeado por objeto

La jerarquía se representa mediante un árbol donde cada nodo almacena:

$$\text{Nodo}_i = \{\text{Contorno}_i, \text{Padre}_i, \text{Hijos}_i\} \quad (2.16)$$

### Aproximación y Simplificación de Contornos

Los contornos detectados suelen contener información redundante. La aproximación poligonal reduce el número de puntos preservando la forma esencial.

#### Algoritmo de Douglas-Peucker

Este método recursivo aproxima una curva mediante segmentos de línea recta:

1. Conectar primer y último punto del contorno
2. Calcular distancia perpendicular de todos los puntos intermedios a esta línea
3. Si  $d_{max} > \epsilon$ , dividir contorno en el punto de máxima distancia

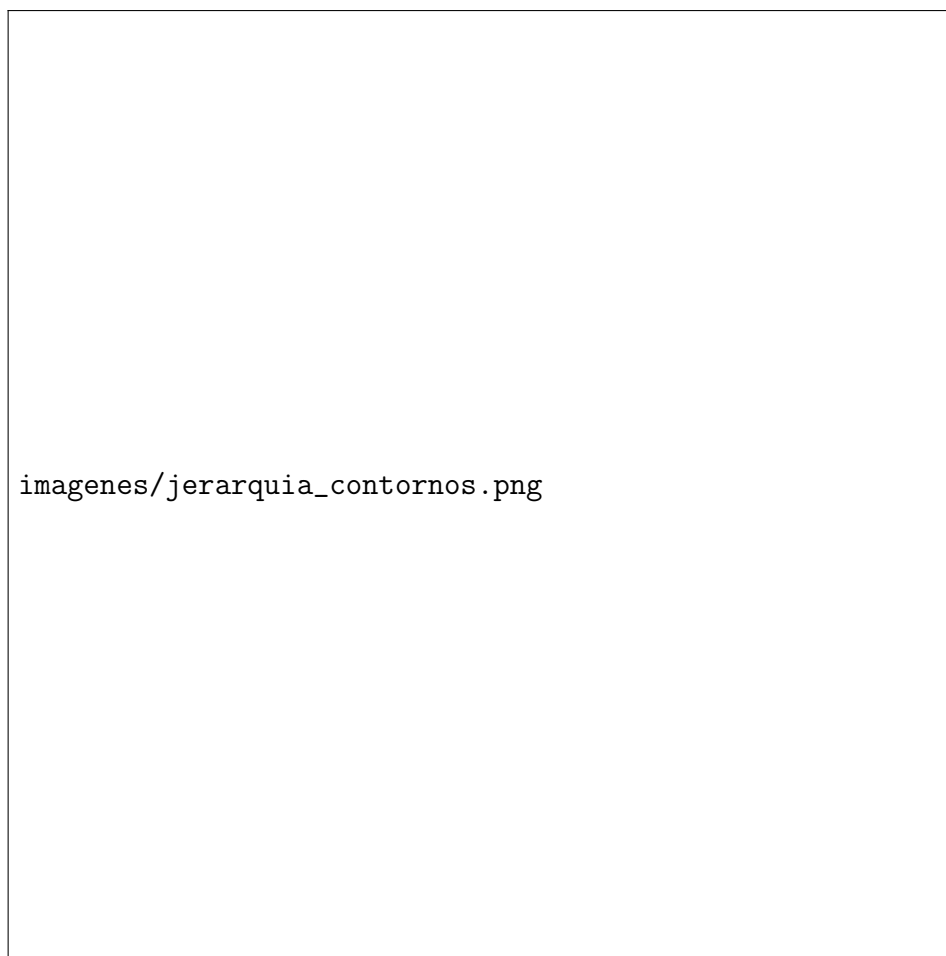


Figura 2.6: Jerarquía de contornos mostrando relación padre-hijo entre contornos externos e internos

4. Aplicar recursivamente a cada segmento
5. Si  $d_{max} \leq \epsilon$ , aproximar segmento con línea recta

La tolerancia  $\epsilon$  controla el nivel de simplificación:

$$\epsilon_{optimal} = \alpha \cdot P_{contorno} \quad (2.17)$$

donde  $P_{contorno}$  es el perímetro y  $\alpha \in [0,01, 0,05]$  es un factor empírico.

## Descriptores Geométricos de Contornos

Los descriptores cuantifican propiedades estructurales de los contornos, permitiendo clasificación y reconocimiento de formas.

### Área

El área encerrada por un contorno se calcula mediante la fórmula del Shoelace:

$$A = \frac{1}{2} \left| \sum_{i=0}^{n-1} (x_i y_{i+1} - x_{i+1} y_i) \right| \quad (2.18)$$

donde  $(x_n, y_n) = (x_0, y_0)$  para cerrar el contorno.

### Perímetro

La longitud del contorno se aproxima sumando distancias euclidianas entre puntos consecutivos:



Figura 2.7: Aproximación poligonal de contorno mediante algoritmo Douglas-Peucker

$$P = \sum_{i=0}^{n-1} \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} \quad (2.19)$$

### Centroide

El centro de masa del contorno se calcula como:

$$\bar{x} = \frac{1}{6A} \sum_{i=0}^{n-1} (x_i + x_{i+1})(x_i y_{i+1} - x_{i+1} y_i) \quad (2.20)$$

$$\bar{y} = \frac{1}{6A} \sum_{i=0}^{n-1} (y_i + y_{i+1})(x_i y_{i+1} - x_{i+1} y_i) \quad (2.21)$$

### Momentos de Imagen

Los momentos proporcionan información sobre la distribución espacial del contorno:

$$M_{pq} = \sum_{i=0}^{n-1} x_i^p y_i^q \quad (2.22)$$

Los momentos centrales, invariantes a traslación, se definen como:

## Proyecto Final de Estudios: Robot cosechador automático

---

$$\mu_{pq} = \sum_{i=0}^{n-1} (x_i - \bar{x})^p (y_i - \bar{y})^q \quad (2.23)$$

### Razón de Aspecto

La relación entre dimensiones principales se calcula mediante el rectángulo delimitador mínimo:

$$AR = \frac{W_{bbox}}{H_{bbox}} \quad (2.24)$$

donde  $W_{bbox}$  y  $H_{bbox}$  son ancho y alto del bounding box.

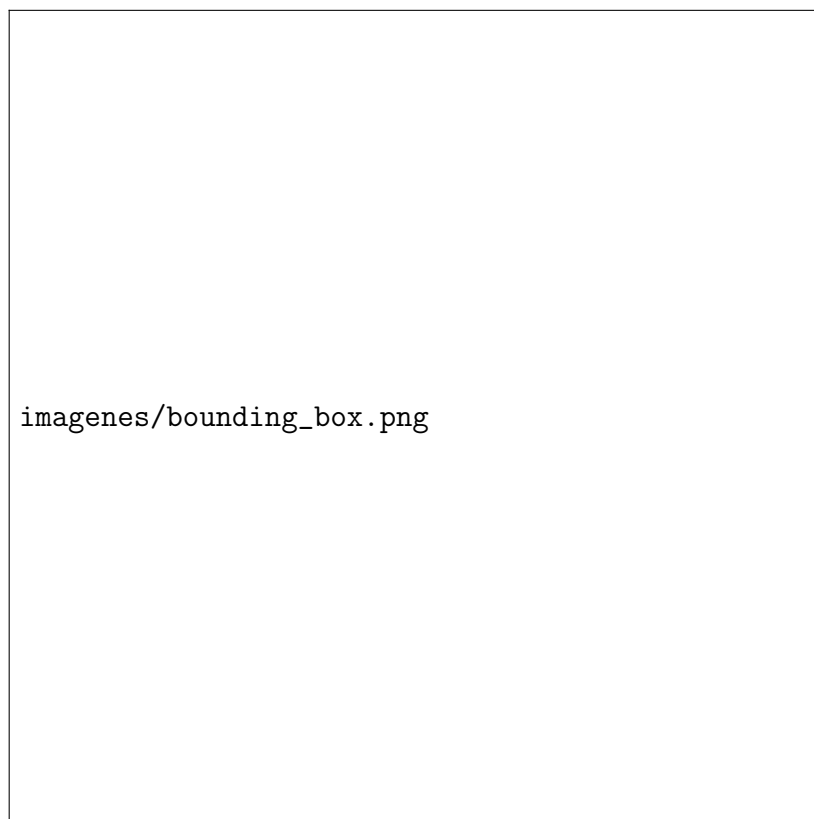


Figura 2.8: Rectángulo delimitador y ejes principales de un contorno

### Circularidad

Mide cuán similar es un contorno a un círculo:

$$C = \frac{4\pi A}{P^2} \quad (2.25)$$

Propiedades:

- $C = 1$  para círculos perfectos
- $C < 1$  para formas no circulares
- $C \rightarrow 0$  para formas muy irregulares

### Convexidad

La relación entre el área del contorno y su envolvente convexa:

$$\text{Convexity} = \frac{A_{convexHull}}{A_{contour}} \quad (2.26)$$

## Proyecto Final de Estudios: Robot cosechador automático

Valores cercanos a 1 indican formas convexas; valores mayores indican concavidades.

### Solidez

Mide la proporción del contorno que ocupa su envolvente convexa:

$$\text{Solidity} = \frac{A_{\text{contour}}}{A_{\text{convexHull}}} \quad (2.27)$$

## Filtrado de Contornos

En aplicaciones reales, las imágenes contienen múltiples contornos, muchos correspondientes a ruido. El filtrado elimina contornos irrelevantes según criterios definidos.

### Filtrado por Área

Eliminar contornos cuya área esté fuera de un rango esperado:

$$\text{Contorno válido} \Leftrightarrow A_{\min} \leq A_{\text{contour}} \leq A_{\max} \quad (2.28)$$

Los umbrales se establecen según el tamaño esperado de objetos de interés.

### Filtrado por Jerarquía

Seleccionar solo contornos de nivel específico en la jerarquía:

- Nivel 0: Contornos externos principales
- Nivel 1: Huecos dentro de objetos
- Niveles superiores: Estructuras anidadas

### Filtrado por Forma

Utilizar descriptores geométricos como criterio:

$$\text{Validez} = \begin{cases} \text{True} & \text{si } C > C_{\min} \wedge AR \in [AR_{\min}, AR_{\max}] \\ \text{False} & \text{en caso contrario} \end{cases} \quad (2.29)$$

## Análisis de Regiones Internas

Una vez identificados contornos relevantes, el análisis del contenido interior proporciona información adicional para clasificación.

### Extracción de Región de Interés (ROI)

Dado un contorno, extraer la región rectangular que lo contiene:

$$\text{ROI} = I[y_{\min} : y_{\max}, x_{\min} : x_{\max}] \quad (2.30)$$

donde  $(x_{\min}, y_{\min})$  y  $(x_{\max}, y_{\max})$  son las coordenadas extremas del contorno.

### Máscara de Contorno

Crear una máscara binaria donde píxeles internos al contorno valen 1:

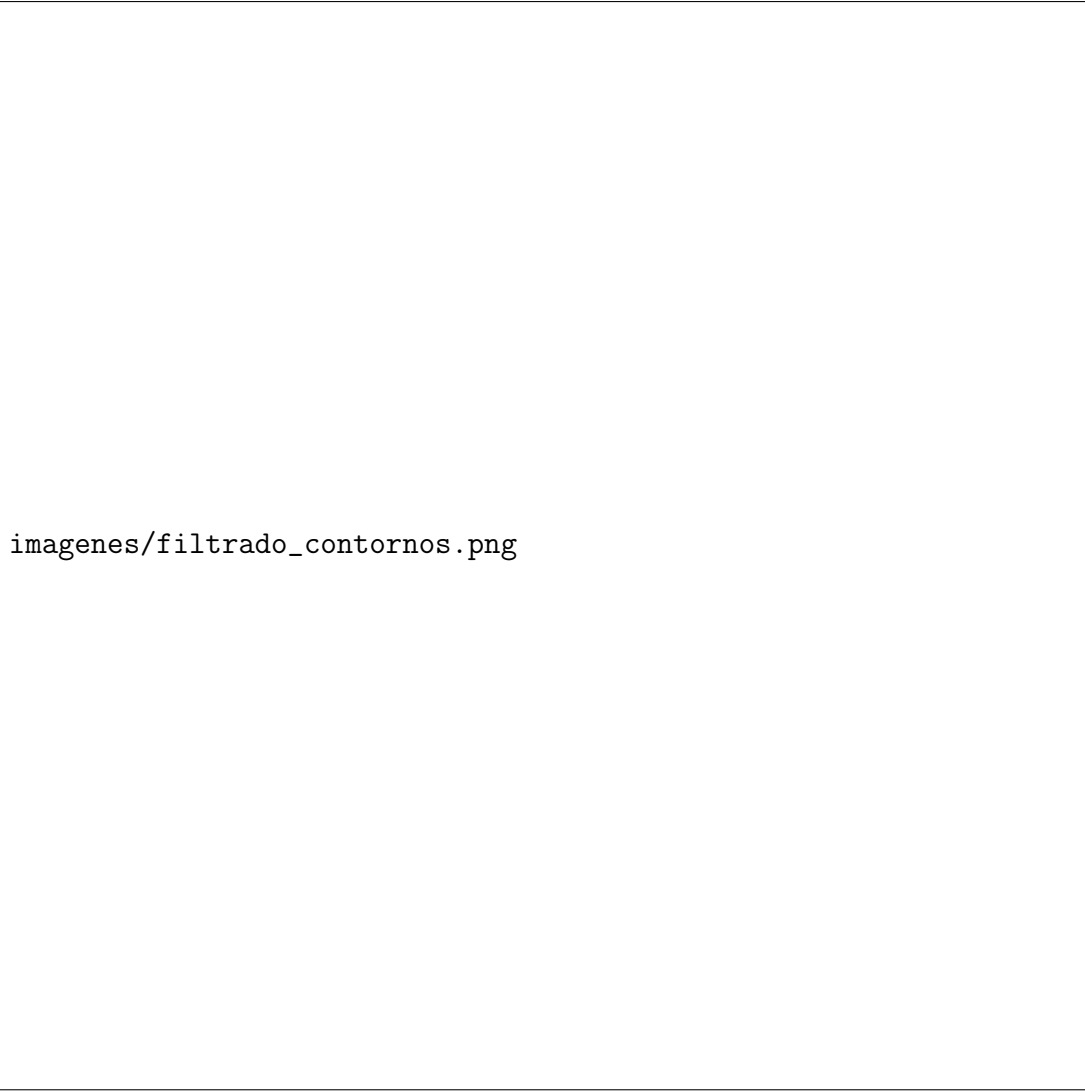
$$M(x, y) = \begin{cases} 1 & \text{si } (x, y) \in \text{Interior}(C) \\ 0 & \text{en caso contrario} \end{cases} \quad (2.31)$$

Esta máscara permite análisis estadístico de píxeles contenidos.

### Conteo de Píxeles

El número de píxeles dentro del contorno proporciona una medida de área alternativa:

$$N_{\text{pixels}} = \sum_{(x,y) \in \text{ROI}} M(x, y) \quad (2.32)$$



imagenes/filtrado\_contornos.png

Figura 2.9: Proceso de filtrado de contornos por área y forma

Esta métrica es menos sensible a irregularidades del perímetro que el cálculo mediante coordenadas del contorno.

## Invariancia a Transformaciones

Para robustez ante variaciones de escala, rotación y traslación, se emplean descriptores invariantes.

### Momentos de Hu

Los siete momentos invariantes de Hu se construyen a partir de momentos centrales normalizados:

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{1+(p+q)/2}} \quad (2.33)$$

El primer momento de Hu, por ejemplo:

$$h_1 = \eta_{20} + \eta_{02} \quad (2.34)$$

Estos momentos permanecen constantes bajo rotación, escala y traslación.

### Descriptores de Fourier

La transformada de Fourier del contorno proporciona representación en frecuencia invariante a transformaciones:



$$F_n = \sum_{k=0}^{N-1} z_k e^{-i2\pi nk/N} \quad (2.35)$$

donde  $z_k = x_k + iy_k$  es la representación compleja del contorno. Los coeficientes de baja frecuencia capturan la forma global, mientras los de alta frecuencia representan detalles.

### 2.2.3. Procesamiento Digital de Imágenes

El procesamiento digital de imágenes constituye el fundamento tecnológico de los sistemas de visión artificial en robótica agrícola. Esta disciplina integra conceptos de análisis matemático, álgebra lineal y teoría de señales para transformar datos visuales crudos en información estructurada y procesable por algoritmos de decisión.

#### Representación Digital de Imágenes

Una imagen digital se define matemáticamente como una función bidimensional discreta  $f(x, y)$ , donde las coordenadas espaciales  $(x, y)$  representan posiciones en el plano de la imagen y el valor de la función indica la intensidad luminosa en cada punto. En el contexto de imágenes a color en formato RGB, esta representación se extiende a un espacio tridimensional:

$$\mathbf{I}(x, y) = \begin{bmatrix} R(x, y) \\ G(x, y) \\ B(x, y) \end{bmatrix} \in [0, 255]^3 \quad (2.36)$$

donde  $R$ ,  $G$  y  $B$  representan los canales de color rojo, verde y azul respectivamente. Esta representación permite codificar aproximadamente 16.7 millones de colores distintos ( $256^3$ ), cubriendo ampliamente el espectro visible necesario para aplicaciones de agricultura de precisión.

La resolución espacial de una imagen determina la cantidad de información disponible para el análisis. Una imagen de dimensiones  $M \times N$  píxeles contiene  $M \cdot N$  elementos de información, cada uno representando el promedio de la radiancia incidente sobre el área del sensor correspondiente a ese píxel.

#### Espacios de Color y Transformaciones

El espacio de color RGB, aunque intuitivo y ampliamente utilizado en sistemas de captura, presenta limitaciones para ciertas tareas de procesamiento debido a su fuerte correlación entre canales y sensibilidad a variaciones de iluminación. Por esta razón, la conversión a espacios de color alternativos resulta fundamental en visión por computadora.

##### Espacio HSV (Hue, Saturation, Value)

El espacio HSV separa la información cromática (matiz y saturación) de la información de iluminación (valor), proporcionando robustez frente a cambios en las condiciones de luz. La transformación de RGB a HSV se define mediante:

$$V = \max(R, G, B) \quad (2.37)$$

$$S = \begin{cases} \frac{V - \min(R, G, B)}{V} & \text{si } V \neq 0 \\ 0 & \text{si } V = 0 \end{cases} \quad (2.38)$$

$$H = \begin{cases} 60 \cdot \frac{G - B}{V - \min(R, G, B)} & \text{si } V = R \\ 60 \cdot \left( 2 + \frac{B - R}{V - \min(R, G, B)} \right) & \text{si } V = G \\ 60 \cdot \left( 4 + \frac{R - G}{V - \min(R, G, B)} \right) & \text{si } V = B \end{cases} \quad (2.39)$$



Figura 2.10: Representación de una imagen digital en el espacio RGB mostrando los tres canales de color

Esta descomposición resulta particularmente ventajosa para la detección de elementos basada en contraste, ya que el canal  $V$  captura la intensidad luminosa independientemente del color, permitiendo operaciones de umbralización más robustas.

### Ventajas Operativas del Espacio HSV

La separación de la información de brillo en el canal  $V$  permite implementar algoritmos de detección invariantes a cambios de iluminación ambiental. En aplicaciones agrícolas donde las condiciones lumínicas varían significativamente durante el día, esta propiedad resulta crítica para mantener la consistencia operativa del sistema.

Matemáticamente, el canal  $V$  satisface la propiedad de invariancia:

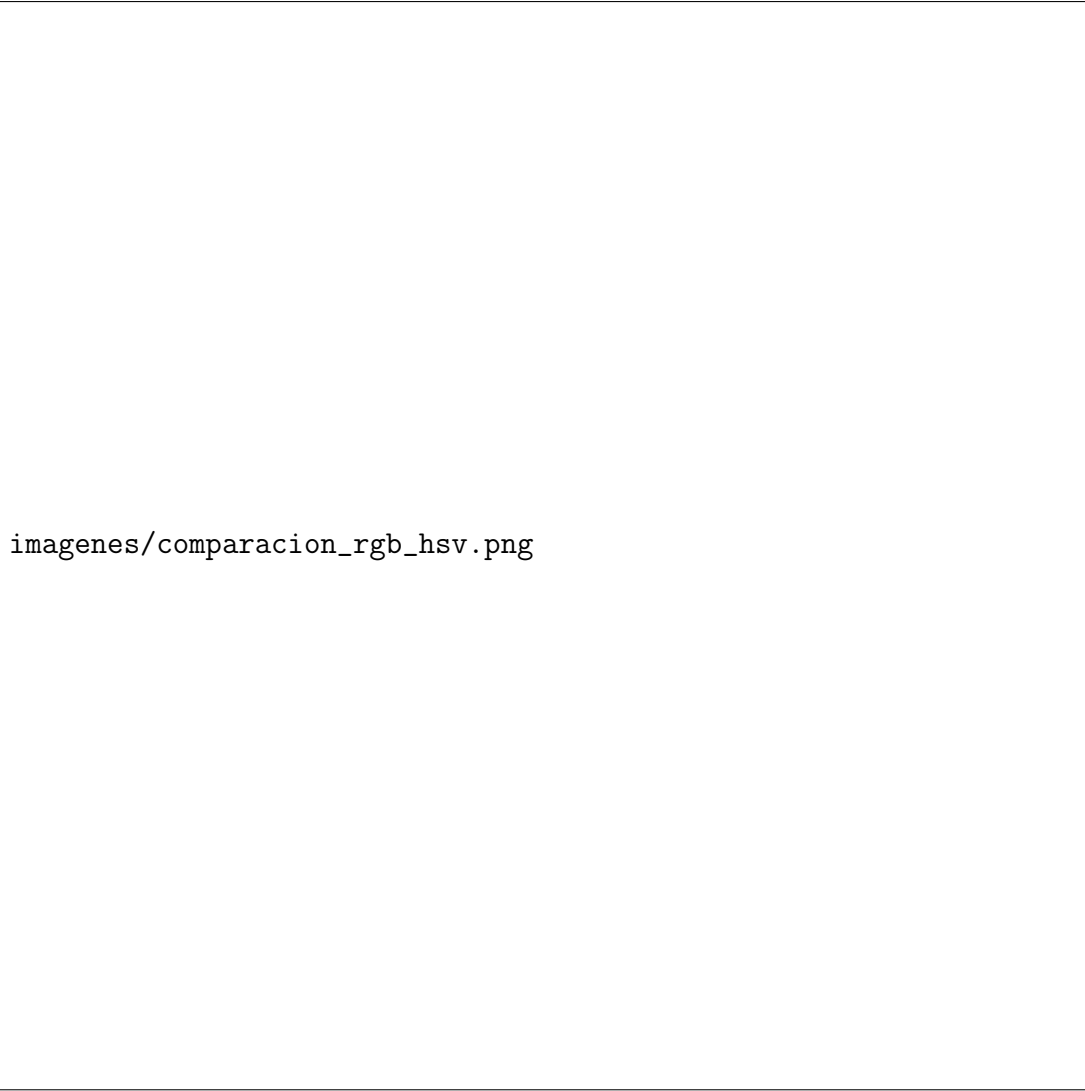
$$V(\alpha \cdot \mathbf{I}_{RGB}) = \alpha \cdot V(\mathbf{I}_{RGB}) \quad \forall \alpha > 0 \quad (2.40)$$

donde  $\alpha$  representa un factor de escala de iluminación global. Esta linealidad permite compensar variaciones uniformes de brillo mediante ajustes simples de umbralización.

### Umbralización y Segmentación

La umbralización constituye una técnica fundamental de segmentación que particiona una imagen en regiones de interés mediante la comparación de intensidades de píxeles contra valores de referencia. La umbralización binaria se define formalmente como:

$$g(x, y) = \begin{cases} 255 & \text{si } f(x, y) > T \\ 0 & \text{si } f(x, y) \leq T \end{cases} \quad (2.41)$$



imagenes/comparacion\_rgb\_hsv.png

Figura 2.11: Comparación entre espacios de color RGB y HSV para una misma escena de cultivo

donde  $T$  es el valor umbral y  $g(x, y)$  es la imagen resultante.

### Umbralización Inversa

En escenarios donde los elementos de interés presentan intensidades menores que el fondo, la umbralización inversa resulta más apropiada:

$$g(x, y) = \begin{cases} 255 & \text{si } f(x, y) < T \\ 0 & \text{si } f(x, y) \geq T \end{cases} \quad (2.42)$$

Esta variante permite destacar regiones oscuras contra fondos claros, técnica ampliamente aplicada en la detección de marcadores de contraste en ambientes controlados.

### Selección Óptima del Umbral

La determinación del valor umbral óptimo  $T^*$  puede realizarse mediante análisis del histograma de intensidades. El método de Otsu minimiza la varianza intra-clase de los grupos resultantes:

$$T^* = \arg \min_T [\omega_0(T)\sigma_0^2(T) + \omega_1(T)\sigma_1^2(T)] \quad (2.43)$$

donde  $\omega_i$  y  $\sigma_i^2$  representan el peso y varianza de cada clase. Alternativamente, puede emplearse conocimiento previo del entorno para establecer umbrales fijos que aprovechen características específicas del escenario operativo.



Figura 2.12: Proceso de umbralización binaria inversa para destacar elementos oscuros

## Operaciones Morfológicas

Las operaciones morfológicas manipulan la forma y estructura de objetos en imágenes binarias mediante la aplicación de elementos estructurantes. Estas operaciones se fundamentan en la teoría de conjuntos y permiten eliminar ruido, rellenar huecos y separar objetos conectados.

### Erosión

La erosión reduce el tamaño de los objetos eliminando píxeles del perímetro:

$$(A \ominus B)(x, y) = \min_{(s,t) \in B} \{A(x + s, y + t)\} \quad (2.44)$$

donde  $A$  es la imagen binaria y  $B$  es el elemento estructurante.

### Dilatación

La dilatación expande los objetos añadiendo píxeles al perímetro:

$$(A \oplus B)(x, y) = \max_{(s,t) \in B} \{A(x + s, y + t)\} \quad (2.45)$$

### Apertura y Cierre

La apertura ( $A \circ B = (A \ominus B) \oplus B$ ) elimina pequeños objetos y protuberancias, mientras el cierre ( $A \bullet B = (A \oplus B) \ominus B$ ) rellena huecos pequeños y conecta componentes próximos. Estas operaciones compuestas resultan esenciales para el refinamiento de máscaras de segmentación en aplicaciones reales.



Figura 2.13: Efecto de operaciones morfológicas sobre una imagen binaria

## Calibración y Transformación de Coordenadas

La calibración establece la correspondencia entre coordenadas píxel en la imagen y coordenadas físicas en el espacio de trabajo. Esta transformación es fundamental para convertir mediciones visuales en comandos de movimiento precisos.

### Modelo de Transformación Lineal

Para configuraciones donde la cámara observa perpendicular al plano de trabajo, la transformación píxel-a-métrica puede aproximarse mediante un modelo lineal:

$$\begin{bmatrix} x_{mm} \\ y_{mm} \end{bmatrix} = \begin{bmatrix} k_x & 0 \\ 0 & k_y \end{bmatrix} \begin{bmatrix} x_{px} \\ y_{px} \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \quad (2.46)$$

donde  $k_x, k_y$  son factores de escala (mm/píxel) y  $t_x, t_y$  son desplazamientos de origen.

### Estimación por Mínimos Cuadrados

Los parámetros de calibración se estiman mediante regresión lineal a partir de puntos de correspondencia conocidos:

$$\mathbf{K} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y} \quad (2.47)$$

donde  $\mathbf{X}$  contiene las coordenadas píxel de puntos de referencia,  $\mathbf{y}$  sus coordenadas métricas medidas físicamente, y  $\mathbf{K}$  el vector de parámetros de calibración. Este método minimiza el error

cuadrático medio garantizando la solución óptima en el sentido de mínimos cuadrados.



Figura 2.14: Proceso de calibración mediante puntos de correspondencia conocidos

### Validación Estadística

La precisión de la calibración se cuantifica mediante el error cuadrático medio (RMSE):

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2} \quad (2.48)$$

donde  $\hat{y}_i$  son las predicciones del modelo. Valores de RMSE inferiores a 1mm resultan aceptables para aplicaciones de manipulación agrícola de precisión.

### 2.2.4. Métricas de Evaluación para Sistemas de Clasificación

La evaluación cuantitativa del desempeño de sistemas de clasificación requiere métricas robustas que capturen diferentes aspectos de la calidad predictiva. En aplicaciones de agricultura de precisión, donde decisiones incorrectas pueden impactar la productividad del cultivo, resulta fundamental comprender las fortalezas y limitaciones de cada métrica.

#### Matriz de Confusión

La matriz de confusión constituye la representación fundamental del desempeño de un clasificador, tabulando predicciones versus etiquetas verdaderas. Para un problema de  $C$  clases, la matriz de confusión  $\mathbf{M} \in \mathbb{R}^{C \times C}$  se define como:

## Proyecto Final de Estudios: Robot cosechador automático

$$M_{ij} = \sum_{k=1}^N \mathbb{I}[y_k = i \wedge \hat{y}_k = j] \quad (2.49)$$

donde  $y_k$  es la clase verdadera de la muestra  $k$ ,  $\hat{y}_k$  es la clase predicha, y  $\mathbb{I}[\cdot]$  es la función indicadora.

Para clasificación binaria, la matriz se simplifica a:

$$\mathbf{M} = \begin{bmatrix} \text{TN} & \text{FP} \\ \text{FN} & \text{TP} \end{bmatrix} \quad (2.50)$$

donde:

- **TP (True Positives)**: Casos positivos correctamente clasificados
- **TN (True Negatives)**: Casos negativos correctamente clasificados
- **FP (False Positives)**: Casos negativos incorrectamente clasificados como positivos (Error Tipo I)
- **FN (False Negatives)**: Casos positivos incorrectamente clasificados como negativos (Error Tipo II)

imagenes/matriz\_confusion\_ejemplo.png

Figura 2.15: Matriz de confusión para clasificación multiclase mostrando patrones de error

### Exactitud (Accuracy)

La exactitud mide la proporción global de predicciones correctas:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} = \frac{\sum_{i=1}^C M_{ii}}{N} \quad (2.51)$$

## Proyecto Final de Estudios: Robot cosechador automático

---

### Ventajas:

- Interpretación intuitiva: porcentaje de aciertos
- Métrica estándar para comparación entre modelos
- Apropiaada cuando todas las clases tienen igual importancia

### Limitaciones Críticas

La exactitud resulta engañosa en datasets desbalanceados. Considere un clasificador que siempre predice la clase mayoritaria en un dataset con 95 % de negativos:

$$\text{Accuracy}_{naive} = \frac{0,95N}{N} = 0,95 \quad (2.52)$$

Este clasificador trivial alcanza 95 % de exactitud sin aprender ningún patrón útil. En agricultura, donde ciertas condiciones pueden ser raras pero críticas, la exactitud puede ocultar fallos graves del modelo.

### Precisión (Precision)

La precisión cuantifica la proporción de predicciones positivas que son correctas:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (2.53)$$

**Interpretación:** De todas las instancias que el modelo clasificó como positivas, ¿qué fracción realmente lo era?

Para clasificación multiclase, se calcula precisión por clase:

$$\text{Precision}_i = \frac{M_{ii}}{\sum_{j=1}^C M_{ji}} \quad (2.54)$$

La precisión es crucial cuando el costo de falsos positivos es alto. En cosecha automatizada, alta precisión garantiza que el robot no intente cosechar elementos no deseados.

### Exhaustividad (Recall/Sensitivity)

El recall mide la proporción de casos positivos reales que fueron correctamente identificados:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (2.55)$$

**Interpretación:** De todas las instancias que realmente son positivas, ¿qué fracción detectó el modelo?

Para clasificación multiclase:

$$\text{Recall}_i = \frac{M_{ii}}{\sum_{j=1}^C M_{ij}} \quad (2.56)$$

El recall es fundamental cuando el costo de falsos negativos es alto. En detección de elementos para cosecha, alto recall asegura que no se pierdan productos maduros.



## Proyecto Final de Estudios: Robot cosechador automático

---

### F-Score

El F-Score armoniza precisión y recall mediante su media armónica:

$$F_1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} = \frac{2TP}{2TP + FP + FN} \quad (2.57)$$

### Generalización F-Beta

Cuando se desea ponderar diferentemente precisión y recall:

$$F_\beta = (1 + \beta^2) \cdot \frac{\text{Precision} \cdot \text{Recall}}{\beta^2 \cdot \text{Precision} + \text{Recall}} \quad (2.58)$$

- $\beta < 1$ : Favorece precisión
- $\beta = 1$ : Balance equitativo (F1-Score)
- $\beta > 1$ : Favorece recall



Figura 2.16: Trade-off entre precisión y recall para diferentes valores de umbral

### Métricas Agregadas Multiclase

Para problemas con  $C > 2$  clases, las métricas por clase se agregan mediante:

#### Macro-Average

Promedio simple de métricas por clase:

## Proyecto Final de Estudios: Robot cosechador automático

---

$$\text{Metric}_{macro} = \frac{1}{C} \sum_{i=1}^C \text{Metric}_i \quad (2.59)$$

Trata todas las clases por igual, apropiado cuando cada clase tiene igual importancia independientemente de su frecuencia.

### Weighted-Average

Promedio ponderado por soporte de cada clase:

$$\text{Metric}_{weighted} = \frac{1}{N} \sum_{i=1}^C n_i \cdot \text{Metric}_i \quad (2.60)$$

donde  $n_i = \sum_{j=1}^C M_{ij}$  es el número de instancias verdaderas de clase  $i$ .

### Micro-Average

Calcula métricas globales agregando TP, FP, FN de todas las clases:

$$\text{Precision}_{micro} = \frac{\sum_{i=1}^C \text{TP}_i}{\sum_{i=1}^C (\text{TP}_i + \text{FP}_i)} \quad (2.61)$$

## Métricas de Error Absoluto

Para clasificadores basados en descriptores continuos, es relevante evaluar el error en la estimación del descriptor mismo.

### Error Absoluto Medio (MAE)

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i| \quad (2.62)$$

Mide la magnitud promedio de los errores sin considerar su dirección.

### Error Cuadrático Medio (RMSE)

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2} \quad (2.63)$$

Penaliza errores grandes más fuertemente que MAE debido a la elevación al cuadrado.

### Error Porcentual Absoluto Medio (MAPE)

$$\text{MAPE} = \frac{100\%}{N} \sum_{i=1}^N \left| \frac{y_i - \hat{y}_i}{y_i} \right| \quad (2.64)$$

Expresa el error como porcentaje del valor verdadero, útil para comparar desempeño en magnitudes diferentes.


## Coefficiente de Determinación

Para regresión de descriptores continuos, el coeficiente  $R^2$  mide la proporción de varianza explicada:

$$R^2 = 1 - \frac{\sum_{i=1}^N (y_i - \hat{y}_i)^2}{\sum_{i=1}^N (y_i - \bar{y})^2} \quad (2.65)$$

donde  $\bar{y}$  es la media de los valores observados.

### Interpretación:



imagenes/distribucion\_errores.png

Figura 2.17: Distribución de errores absolutos en la estimación de descriptores

- $R^2 = 1$ : Predicción perfecta
- $R^2 = 0$ : Modelo no mejor que predecir la media
- $R^2 < 0$ : Modelo peor que predecir la media

## Métricas de Confiabilidad

### Intervalo de Confianza

Para una métrica estimada  $\hat{\theta}$  con error estándar  $SE$ , el intervalo de confianza al  $(1 - \alpha)100\%$  es:

$$IC_{1-\alpha} = \hat{\theta} \pm z_{\alpha/2} \cdot SE \quad (2.66)$$

Para exactitud:

$$SE_{Acc} = \sqrt{\frac{\hat{p}(1 - \hat{p})}{n}} \quad (2.67)$$

### Coefficiente de Variación

Mide la variabilidad relativa de una métrica:

$$CV = \frac{\sigma}{\mu} \times 100\% \quad (2.68)$$

Valores bajos de CV indican consistencia del clasificador entre diferentes conjuntos de prueba.

## Proyecto Final de Estudios: Robot cosechador automático

### Métricas de Robustez

#### Estabilidad ante Perturbaciones

Evalúa degradación de desempeño bajo ruido:

$$\Delta\text{Perf} = \frac{\text{Perf}_{\text{clean}} - \text{Perf}_{\text{noisy}}}{\text{Perf}_{\text{clean}}} \times 100 \% \quad (2.69)$$


Sistemas robustos exhiben  $\Delta\text{Perf} < 10 \%$  para niveles moderados de ruido.

#### Repetibilidad

Desviación estándar de la métrica en ejecuciones repetidas:

$$\sigma_{\text{rep}} = \sqrt{\frac{1}{K-1} \sum_{k=1}^K (\text{Metric}_k - \bar{\text{Metric}})^2} \quad (2.70)$$

donde  $K$  es el número de repeticiones.



imagenes/robustez\_ruido.png

Figura 2.18: Degradación de exactitud con incremento de ruido gaussiano

### Selección de Métricas según Aplicación

Para sistemas de clasificación en agricultura de precisión, la selección de métricas debe alinearse con los objetivos operativos:

#### Prioridades Típicas:

1. **Alta Exhaustividad:** Minimizar productos no detectados

2. **Precisión Aceptable:** Limitar clasificaciones incorrectas
3. **Robustez:** Mantener desempeño bajo variaciones ambientales
4. **Consistencia:** Bajo coeficiente de variación entre sesiones

La métrica compuesta óptima puede definirse como:

$$\text{Score}_{total} = w_1 \cdot F_\beta + w_2 \cdot (1 - CV) + w_3 \cdot (1 - \Delta\text{Perf}) \quad (2.71)$$

donde  $w_i$  son pesos que reflejan la importancia relativa de cada aspecto, con  $\sum w_i = 1$ .

## 2.3. Cinemática de Robots Cartesianos

## 2.4. Sistemas de Transmisión Mecánica

## 2.5. Control de Motores Paso a Paso



## **3. Desarrollo del Sistema**

### **3.1. Arquitectura General del Sistema**

### **3.2. Modelado y Diseño Mecánico**

#### **3.2.1. Especificaciones de Diseño y Restricciones**

#### **3.2.2. Análisis Cinemático del Sistema**

#### **3.2.3. Diseño Estructural**

#### **3.2.4. Sistema de Movimiento Horizontal**

#### **3.2.5. Sistema de Movimiento Vertical**

#### **3.2.6. Brazo Robótico**

#### **3.2.7. Modelado CAD y Fabricación**

### **3.3. Sistema de Control de Bajo Nivel (Nivel Regulatorio)**

#### **3.3.1. Arquitectura del Nivel Regulatorio**

#### **3.3.2. Hardware de Control - Arduino Mega 2560**

#### **3.3.3. Selección y Dimensionamiento de Actuadores**

#### **3.3.4. Sensores de Seguridad**

#### **3.3.5. Control de Movimiento**

#### **3.3.6. Protocolo de Comunicación UART**

### **3.4. Sistema de Supervisión y Alta Gestión (Nivel Supervisor)**

#### **3.4.1. Arquitectura del Nivel Supervisor**

#### **3.4.2. Hardware Supervisor - Raspberry Pi 4**

#### **3.4.3. Máquina de Estados Supervisora**

#### **3.4.4. Coordinación con Nivel Regulatorio**

### **3.5. Inteligencia Artificial y Visión por Computadora**

#### **3.5.1. Arquitectura General del Sistema de IA**

#### **3.5.2. Sistema de Visión Artificial**

#### **3.5.3. Calibración Espacial del Sistema**

## **4. Pruebas y Resultados**

### **4.1. Metodología de Pruebas**

### **4.2. Pruebas del Sistema Mecánico**

### **4.3. Pruebas del Sistema de Control**

### **4.4. Pruebas del Sistema de IA**

### **4.5. Pruebas de Integración**

### **4.6. Análisis de Resultados**



## **5. Conclusiones y Trabajo Futuro**

### **5.1. Conclusiones Generales**

### **5.2. Aportes del Proyecto**

### **5.3. Trabajo Futuro**

## A. Diagramas Eléctricos Completos

## B. Código Fuente Relevante

## C. Especificaciones Técnicas de Componentes

## D. Manual de Usuario

## E. Hojas de Datos

# Referencias