



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

Máster Universitario de Automática e
Informática Industrial

Universidad Politécnica de Valencia

**Reconstrucción 3D de bayas de racimos de
uva de vinificación basada en detección de
elipses sobre imágenes de estéreo-visión**

Tesina de Master

Autor: Carlos Alberto Molina Toscano

Director: Antonio José Sánchez Salmerón

Agradecimientos

Siempre resulta difícil agradecer a aquellas personas que han colaborado con un proceso, con un trabajo, porque nunca alcanza el tiempo, el papel o la memoria para mencionar o dar con justicia todos los créditos y méritos a quienes se lo merecen.

Partiendo de esta limitación y diciendo de antemano MUCHAS GRACIAS a todas las personas que de una u otra manera han colaborado en el desarrollo de esta Tesina de Máster, deseo agradecer especialmente a:

Al Instituto de Automática e Informática Industrial (AI2) por haberme brindado todas las facilidades tanto técnicas como humanas, y sobre todo a Antonio Sánchez, mi director de proyecto, por todo el seguimiento, control y tiempo que me ha dedicado, además de la paciencia que ha tenido conmigo.

Y, principalmente, agradecer a toda mi familia por la paciencia y apoyo moral que en todo momento han sabido transmitirme, no hay palabras suficientes para demostrar mi gratitud, simplemente me queda decirles GRACIAS POR ESTAR SIEMPRE ALLÍ.

Resumen

Este trabajo recoge un procedimiento para la estimación del tamaño de bayas de uva, que es considerado como un parámetro que permite valorar la calidad de los racimos de uva que se utilizan en la elaboración de vinos, ya que, establece una relación entre la piel y la pulpa.

Para la consecución del objetivo a lo largo del trabajo se tratará la problemática de la detección de elipses respecto del caso más simple, que es la detección de círculos, pasando por el procesamiento de imagen para realzar las características a detectar, que en este caso serán los bordes de bayas de uva.

La detección se hace sobre un par de imágenes estéreo, capturadas con una cámara dotada con sistema estereoscópico, por lo que existirán para cada una de las bayas detectables una elipse (o círculo) izquierdo y derecho. Para lograr que las parejas de elipses correspondan a una misma baya se propone seguir un algoritmo de emparejamiento, que se basa en las características geométricas y de aspecto de las bayas.

Una vez que se obtienen las elipses, que son la representación 2D del objeto 3D (las bayas de uva), es decir, son las proyecciones sobre la imagen con las que se consigue un modelo representativo de las bayas en 3D, de donde, finalmente se estima el tamaño medio de baya (radio).

Motivación

En mi vida estudiantil siempre me he orientado en aprender o hacer énfasis en nuevos conocimientos, con el fin de abrir una nueva puerta a las posibilidades que se pudieran dar en el futuro. En ese sentido, el área de visión que se abordaba durante el máster dio pie a que se despertara en mí el interés por saber más acerca de éste área.

En concreto, la realización de esta Tesina de Máster me ha dado la posibilidad de ahondar en un tema del que poseía pocos conocimientos. Aunque desde un enfoque más bien orientado a la investigación, se ha tratado un tema muy práctico y que tiene cada vez más importancia en la industria, en este caso la agroalimentaria y más concretamente en la vinícola.

Contenido

1.	Introducción	5
1.1.	Planteamiento del problema	6
1.2.	Objetivos	7
1.3.	Estado del arte	7
2.	Círculos	11
2.1.	Procesado de imágenes.....	11
2.1.1.	Captura de imágenes	12
2.1.2.	Segmentación.....	13
2.1.3.	Detección de bordes	14
2.1.4.	Estimación de círculos.....	16
2.2.	Búsqueda de correspondencias	20
2.3.	Reconstrucción 3D y estimación del tamaño de baya	27
2.3.1.	Triangularización.....	27
2.3.2.	Refinamiento.....	29
2.4.	Resultados	31
3.	Elipses.....	34
3.1.	Algoritmo de detección basado en la curvatura	34
3.1.1.	Restricciones en los bordes.....	35
3.1.2.	Ventajas e inconvenientes	38
3.2.	Algoritmo de detección basado en modelos.....	39
3.2.1.	Desarrollo	39
3.2.2.	Ventajas e inconvenientes	46
3.3.	Algoritmo propuesto	47
3.3.1.	Procesado de imagen	47
3.3.2.	Ventajas e inconvenientes	55
3.4.	Conclusiones de la detección de elipses	57
3.5.	Búsqueda de correspondencias.	58
3.6.	Reconstrucción y estimación de tamaño	60

3.7. Resultados	62
4. Conclusiones finales	64
5. Trabajos futuros	65
6. Anexos	66
6.1. Búsqueda de correspondencias para Elipses	66
6.2. Mejoras en la detección	68
6.3. Círculos vs. Elipses	71
7. Referencias	72

1. Introducción

Los sistemas de visión por computador han demostrado ser, en los últimos años, una herramienta de gran ayuda en la mejora de la eficiencia en la realización de procesos y obtención de productos, reduciendo el error de factor humano, causado por principalmente por la fatiga o las distracciones.

Los sistemas de visión se basaban sobre todo en la utilización de información 2D. Sin embargo, la evolución tecnológica ha hecho que hoy en día se aproveche tanto la información 2D como 3D, lo que expande su uso a diferentes sectores industriales. La tarea pendiente es, por tanto, cómo manejar ésta información de forma que pueda ser útil para cumplir un determinado fin.

Una de las industrias que está sacando provecho de la implementación de los sistemas de visión es la industria agroalimentaria, la cual los utiliza principalmente en tareas de inspección y clasificación del producto. En particular, en este trabajo, se trata con la industria vitivinícola, donde, mediante la manipulación de la información 2D y 3D se pretende desarrollar una herramienta que inspeccione los racimos de uva para determinar su calidad, de forma que sea la mejor uva la que se utilice en la elaboración de vinos.

La calidad del vino depende principalmente de la calidad de las bayas de uva con la que se elabora. Las prácticas actuales para estimar la calidad de la uva requieren costosos análisis destructivos o están sujetos a la subjetividad del factor humano, esto, sumado a la rapidez con la que se necesita obtener resultados durante la cosecha, la falta de herramientas de medida y/o su alto coste, hacen que medir la calidad de la uva sea una tarea complicada.

Uno de los factores que determinan la calidad de la uva es el tamaño, ya que éste establece una relación entre la piel y la pulpa. Recientemente se ha publicado un trabajo de investigación orientado a la estimación del tamaño de bayas de uva de vino [1], donde se detectan círculos sobre una imagen y se clasifican los granos de uva. En este trabajo se estima el tamaño de las bayas de uva mediante un sistema monocular y la inserción de un objeto de calibración de tamaño conocido en la escena. Para evitar el uso de un objeto de calibración en la escena, en la tesina se plantea el uso de un sistema de visión estereoscópico.

Un racimo de uva tiene una geometría compleja formada por un conjunto de cuasi-esferas, que causan ocultamiento entre ellas y por tanto se está perdiendo información, lo que provoca que el control de calidad por visión artificial resulte un problema complejo de resolver. Sin embargo, existen métodos que permiten extraer (estimar) la información 3D a partir de imágenes 2D.

Estos métodos se pueden englobar en dos clases: activos y pasivos. Los métodos activos [2] [3] son aquellos que emiten un patrón de luz sobre la escena para analizar su deformación, en tanto que los métodos pasivos solo utilizan la iluminación ambiente, como ocurre con la estereoscopia [4].

Aunque la estereoscopia es muy simple, en particular en la inspección de alimentos no está muy extendida. Esto se debe a que, principalmente, el problema de la correspondencia de las imágenes estereó es difícil de resolver, pues encontrar características visuales que aparezcan en el par de imágenes no es una tarea trivial. Los trabajos que existen y que proponen una solución están basados principalmente en la detección de bordes.

A partir de los bordes se estiman formas como elipses o círculos, dicha estimación es uno de los problemas que desde hace años viene siendo estudiada y que aún sigue sin tener una solución generalista. Éste problema, de solución compleja, es abordado siguiendo diferentes vertientes de investigación. Entre los métodos de resolución están los basados en la transformada Hough, mínimos cuadrados, algoritmos genéticos e híbridos de estos. De los cuales los más ampliamente difundidos son los dos primeros.

Transformada Hough

Para su implementación en la detección de elipses requiere de la estimación de 5 parámetros (posición (X, Y) del centro, eje mayor, eje menor y orientación), en contraposición con los 2 parámetros, en coordenadas polares (ρ, θ) , que definen las líneas rectas; lo que conlleva un elevado coste computacional. Es por ello que muchas veces las investigaciones se centran en la reducción de la dimensionalidad, lo que implicaría, a priori, una reducción de los tiempos de cómputo. Por lo tanto, siempre que sea factible, se prefiere estimar círculos (definidos por 3 parámetros) que elipses.

Mínimos cuadrados (Least Square Mean)

Requiere la resolución de ecuaciones matriciales, a partir de las cuales se pretende minimizar una función de coste o una distancia algebraica. Se suelen probar para que sean lineales y se facilite su resolución, con lo que se reduce el tiempo de cómputo, haciendo de éste método muy adecuado en aplicaciones de tiempo real. También está su variante para trabajar de forma no lineal, cuyos inconvenientes radican en su sensibilidad a datos anómalos y que sólo se puede calcular una primitiva a la vez.

1.1. Planteamiento del problema

Se pretende determinar de forma automática el tamaño medio de las bayas de uva de un racimo para cualificar su calidad, además de realizar una reconstrucción en 3D del mismo. Para ello, como se ha mencionado anteriormente, uno de los problemas a resolver es el de las correspondencias.

Se plantea resolver el problema por medio de un procedimiento estereoscópico, buscando correspondencias entre las elipses¹ detectadas en la imagen izquierda y las elipses

¹ Dada la geometría de las bayas de uvas, forma cuasi-esférica, la detección de elipses puede ser simplificada a una detección de círculos.

detectadas en la imagen derecha. La detección se realiza utilizando un algoritmo de detección, como el Algoritmo Hough [5], e incluyendo restricciones epipolares, de gradientes y de color.

Una vez encontradas las elipses que se corresponden, se reconstruye el racimo de uvas en 3D, desde el cual se estima el tamaño medio de las bayas.

1.2. Objetivos

Una vez se ha planteado el problema a resolver, en el presente trabajo se define como objetivo principal la estimación del tamaño del racimo de uva, con el que se establece un proceso automático de clasificación de racimos de uva, por tamaño, mediante el uso de técnicas de visión, 2D y 3D.

Para cumplir con el objetivo, se divide el proceso en 3 fases:

- Procesado de imágenes
- Búsqueda de correspondencias
- Reconstrucción 3D y estimación de tamaño.

Todo el proceso se lleva a cabo mediante el software informático MATLAB.

1.3. Estado del arte

El problema de la detección de elipses se lleva estudiando desde hace varios años, es por ello que existe un gran número de propuestas que, con mayor o menor eficiencia, han intentado resolverlo. Todas ellas, según el método de detección en el que están basados, pueden ser encasilladas en uno de estos 4 grupos: Transformada Hough, Least Square Mean, Algoritmos Genéticos e híbridos de estos. Actualmente se está planteando el uso de lógica difusa (fuzzy) o redes neuronales, sobre todo de cara a mejorar la calidad² de los bordes, los cuales son utilizados como punto de partida en la detección (estimación) de elipses.

Un estudio, no excesivamente detallado, pero que sirve de alguna manera como orientación para conocer las propuestas que han sido planteadas durante este tiempo y brindar una idea de hacia dónde están evolucionando las investigaciones sobre la detección de elipses se puede ver en [6].

Son pocas las propuestas que, como la planteada por Prasad y Leung [7] [8], algoritmo basado en el análisis de curvatura o Prasad, han sido probadas sobre imágenes reales, es decir, los bordes utilizados para estimar elipses han sido obtenidos a partir de las imágenes RGB de entornos cualquiera y no generados sintéticamente. Éste hecho es un punto muy a tener en cuenta, ya que los autores utilizan la curvatura de los bordes como característica que define a las elipses a detectar.

² Se considera que un borde es de buena calidad cuando está bien definido, tiene una curvatura suficiente y realmente representa el o parte del contorno de una uva.

Los bordes se obtienen implementando el detector de bordes Canny. De acuerdo a la curvatura de los bordes se conforman grupos que podrían formar una elipse. Los grupos se forman realizando un sondeo de bordes acordes dentro de un área de búsqueda (Figura 1.d), la cual se define para cada borde y su tamaño depende de la curvatura. Cada grupo obtiene un índice de relación, ambos son optimizados por mínimos cuadrados, de forma que los grupos formados tengan el menor índice relación.

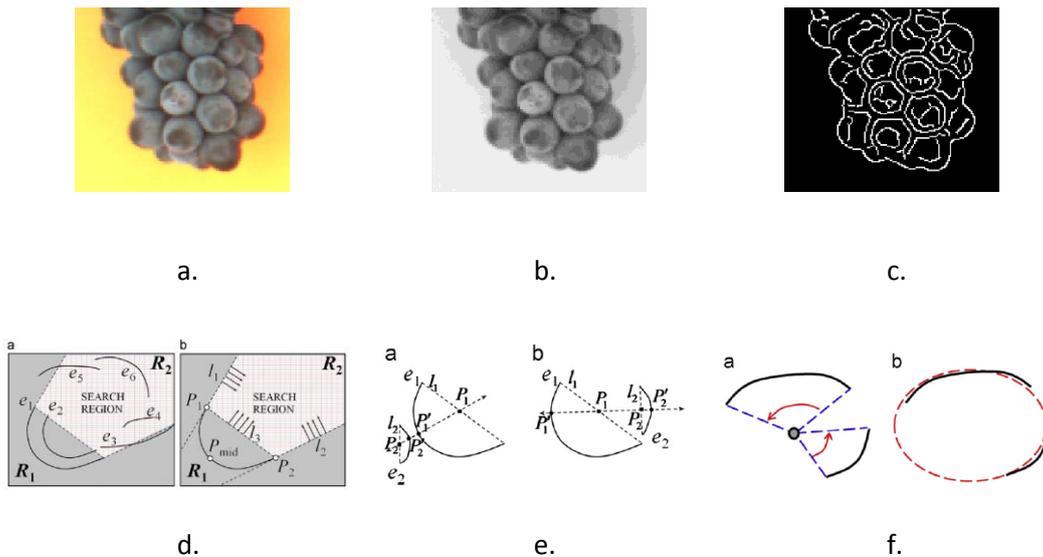


Figura 1. Algoritmo Prasad. a. Imagen RGB de partida. b. Imagen en escala de grises. c. Bordes detectados con Canny. d. Área de búsqueda. e. Agrupamiento de bordes. f. Estimación de elipse.

Las elipses detectadas, resultado de la implementación del Algoritmo Prasad sobre una sección de imagen, como las que se utilizan en este trabajo, se pueden apreciar en la Figura 2. Se observa que el número de elipses detectadas es bajo, y que además, existen elipses que superan con creces la incertidumbre de validez³ de detección. Por lo tanto, el algoritmo no es aplicable directamente. Como parte del desarrollo de este trabajo, más adelante, se propone algunas ideas que permitan que este algoritmo mejore su capacidad de detección.

³ La incertidumbre de validez es subjetiva, se considera una elipse no válida aquella que a simple vista no define, con una cierta tolerancia, ninguna de las bayas del racimo.

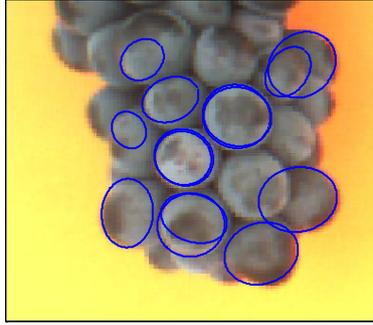


Figura 2. Elipses detectadas con el Algoritmo Prasad.

La propuesta planteada por Liu y otros [9], algoritmo basado en modelos o Algoritmo Pedestrians, está enfocado en la detección de siluetas de personas en cualquier entorno, propone una detección basado en patrones. La obtención del patrón se obtiene mediante un proceso de entrenamiento, en el cual se utilizan imágenes (en escala de grises) del mismo tamaño. Las imágenes son procesadas en busca de características que definan los objetos (gradiente). Por comparación entre todas las imágenes, se obtiene un modelo de probabilidad que define dónde es más probable que esté localizada una persona y un modelo de pesos que define la importancia tiene cada píxel de la imagen para la detección. Éste último es refinado por mínimos cuadrados, resolviendo iterativamente la siguiente ecuación.

$$W^{(k+1)}(x, y) = W^{(k)}(x, y) + \mu \cdot e \cdot P(x, y, X_i(x, y)) \quad (1)$$

donde, x e y son las posiciones horizontal y vertical de cada píxel en la imagen. $W^{(k+1)}(x, y)$ y $W^{(k)}(x, y)$ son los modelos de peso de las iteraciones k -ésima y $k+1$ -ésima en x e y . $P(x, y, X_i(x, y))$ es el modelo de probabilidades definido para una característica X_i en x e y . μ es un parámetro de control de convergencia (en su propuesta los autores lo establecen en 10^{-10}). e es el error.

$$G = \sqrt{G_x^2 + G_y^2} \quad (2)$$

$$\theta = \tan^{-1} \left(\frac{G_y}{G_x} \right) \quad (3)$$

A partir del gradiente de la imagen en escala de grises se definen dos características, magnitud (2) y orientación (3), siendo G_x y G_y el gradiente⁴ en la dirección horizontal y vertical respectivamente; por lo tanto, hay un modelo de probabilidad y de peso tanto para magnitud como para orientación. El clasificador generado a partir de dichas características por separado,

⁴ Se utiliza el detector Sobel.

no es muy robusto, se consideran por tanto clasificadores débiles, combinando ambos se genera un clasificador más robusto, clasificador fuerte, Figura 3.

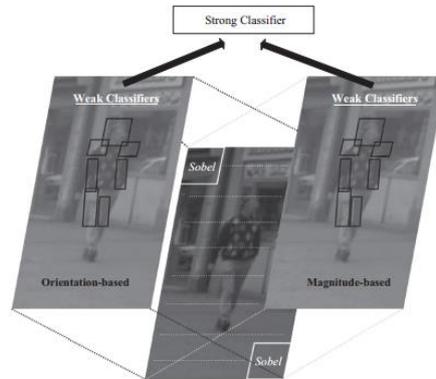


Figura 3. Generación de un clasificador fuerte a partir de los clasificadores débiles de orientación y magnitud del gradiente.

El escenario se recorre con la imagen patrón, buscando zonas que tengan características similares, y por tanto, en ella haya personas. El algoritmo también aplica un escalado del patrón, por tanto, se puede detectar la presencia de personas de distintos tamaños (Figura 4) en los distintos escenarios.



Figura 4. Detección de personas en escenarios no controlados utilizando el Algoritmo Pedestrians

Otros trabajos publicados donde también se busca la detección, pero de objetos con distintas formas o características, son: detección de malezas [10], medición del tamaño del trigo [11], estimación de la firmeza del salmón [12] o la medida del volumen de rodajas de manzana durante el proceso de secado [13].

2. Círculos

La geometría cuasi-esférica de las bayas de uva permite que la detección de elipses sea simplificada por una detección de circunferencias. En este sentido, se propone como solución más sencilla un algoritmo basado en Hough [5].

Las fases en que se divide el proceso de estimación de tamaño de baya, son los mismos que se marcaron como objetivos:

Procesado de imágenes. Se llevan a cabo las operaciones morfológicas necesarias que permitan, en primer lugar, obtener bordes que definan las bayas lo más fielmente posible, y en segundo lugar, utilizarlos para hacer una buena estimación de círculos.

Búsqueda de correspondencias. Se intenta encontrar características geométricas y de color entre los círculos detectados en los pares de imágenes estereoscópicas, que permitan determinar qué elipses corresponden a proyecciones de una misma baya.

Reconstrucción 3D y estimación de tamaño. Se utilizan las proyecciones de bayas que se corresponden para generar un modelo de esferas que componen el racimo de uva, de donde se estima el tamaño promedio de las esferas, que se considerará como el tamaño de baya del racimo.

A continuación se muestra de una forma más detallada cada una de las fases mencionadas.

2.1. Procesado de imágenes

El objetivo del procesado de imágenes, básicamente, es el de resaltar aquellas características que puedan ser de ayuda para la detección de circunferencias que representen las bayas de uva.

Las características a resaltar en esta fase son los bordes de las bayas, buscando que sus contornos sean más relevantes que los contrastes internos, cuanto más representativos sean los bordes, mejor será la detección de circunferencias.

El procesado se ha separado en las siguientes etapas:

- Captura de imágenes.
- Segmentación.
- Detección de bordes
- Estimación de círculos.

2.1.1. Captura de imágenes

Los pares de imágenes RGB, utilizados como información inicial, son capturados con un sistema estereoscópico, Bumblebee2 (Point Grey Research Inc., Richmond BC, Canada) model BB2-08S2C-25 (Figura 5). La captura de las imágenes se ha realizado bajo condiciones de laboratorio con un fondo controlado, buscando que haya un buen contraste entre éste y el racimo de uvas, lo cual permite facilitar el posterior procesamiento. A partir de los datos de calibración [15], las imágenes han sido corregidas de distorsión y rectificadas. En la Figura 6 se muestra un ejemplo de cómo son las imágenes con las que se trabajará en las siguientes etapas.



Figura 5. Sistema de adquisición de imágenes prestado por el IVIA.



a. Izquierda

b. Derecha

Figura 6. Imágenes RGB con distorsión corregida y rectificadas.

Para llevar a cabo el trabajo, se han tomado 40 pares estéreo de imágenes de racimos de las diferentes variedades de uvas.

2.1.2. Segmentación

Tal como se ha mencionado en el apartado anterior, las imágenes han sido tomadas bajo condiciones de laboratorio, forzando a que haya un claro contraste entre fondo⁵ de imagen y el racimo de uvas. De forma que el racimo se puede separar más fácilmente del resto de elementos en la imagen.

La segmentación del racimo del resto de la imagen, como no podía ser de otro modo, está basada en el color. Aprovechando el hecho de que se conoce el color que pueden tener las uvas, se definen umbrales en las capas R, G y B, y de que en la mayoría de casos el racimo de uva es el elemento dominante (mayor superficie) en la imagen, se puede obtener la máscara del racimo.

El valor de los umbrales se ha ido ajustando para diferentes imágenes, de donde se ha obtenido unos parámetros medios de umbral. Sin embargo, dada la diversidad de los racimos, para conseguir una segmentación más aceptable no se toman valores de umbral únicos para todos los racimos, sino que, se separan en 3 grupos según la combinación de colores que hay en la imagen. Así, como se muestra en la Figura 7, cada grupo está formado por imágenes que tienen:

- a. Fondo amarillo y uvas de color azulado oscuro y azulado claro. Son el grupo mayoritario.
- b. Fondo amarillo y uvas de color negruzco y rojizo.
- c. Fondo azul y uvas de color claro variado, que pueden ser azuladas, rojizo y amarillento o verdosas.

En la Figura 8 se muestran ejemplos de máscaras que se consiguen utilizando los criterios mencionados. Como es de esperar de una umbralización promedio, habrá racimos en los que no se logre segmentar tan claramente, debido a las características muy particulares que pueden tener, un par de ejemplos se pueden ver en la Figura 9.



Figura 7. Ejemplos de imágenes con las que se va a tratar.

⁵ Fondo de color amarillo para uvas oscuras y azul para uvas verdosas.

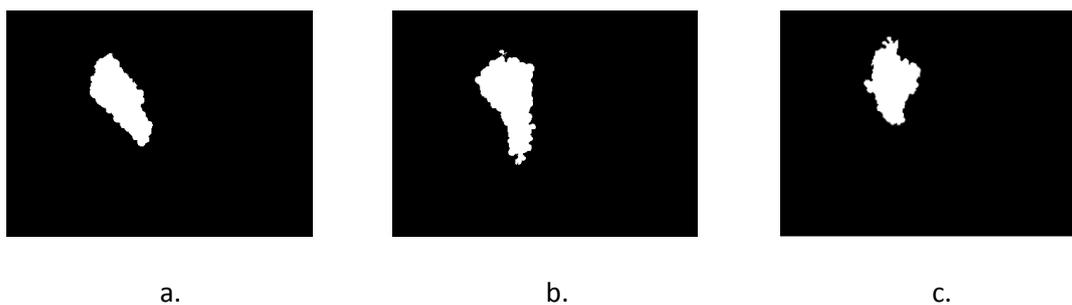


Figura 8. Ejemplos de máscaras de racimos segmentados.



Figura 9. Ejemplos de imágenes con segmentado imperfecto.

Sin embargo, este incorrecto segmentado, como se mostrará más adelante, no presenta mayor inconveniente que el de detectar un mayor número de falsas circunferencias.

2.1.3. Detección de bordes

Al contrario que en la segmentación, donde se utilizan imágenes RGB, la detección de bordes se realiza sobre el equivalente en escala de grises. El utilizar imágenes en escala de grises produce prácticamente los mismos resultados que utilizar cada una de las capas RGB y combinarlas, éste hecho se constató durante las pruebas iniciales realizadas. La gran ventaja del uso de las imágenes en escala de grises es el menor volumen datos (un tercio) que hay que procesar.

Las imágenes en escala de grises se obtienen, de forma sencilla, utilizando la función 'rgb2gray' de MATLAB, que aplica la relación $0.2989 \cdot R + 0.5870 \cdot G + 0.1140 \cdot B$ [16] para obtener la intensidad de cada píxel.



Figura 10. Imagen en escala de grises obtenida con 'rgb2gray' de MATLAB

Sobre la imagen en escala de grises (I) se calcula la variación de intensidad que hay entre los píxeles contiguos, para ello, se utiliza la función 'gradient' de MATLAB (4), donde se devuelve como resultado la variación de intensidad en horizontal (G_x) y en vertical (G_y), siendo los dos parámetros adicionales la forma de especificar que la comparación se hace entre píxeles adjuntos, vertical u horizontalmente.

$$[G_x \ G_y] = \text{gradient}(I, 1, 1) \quad (4)$$

Se obtiene la magnitud del gradiente (G) a partir de sus componentes utilizando (2), en la siguiente figura se muestra el resultado, donde, además, se ha utilizado la máscara del racimo obtenida durante la segmentación para que solo quede la información realmente útil.

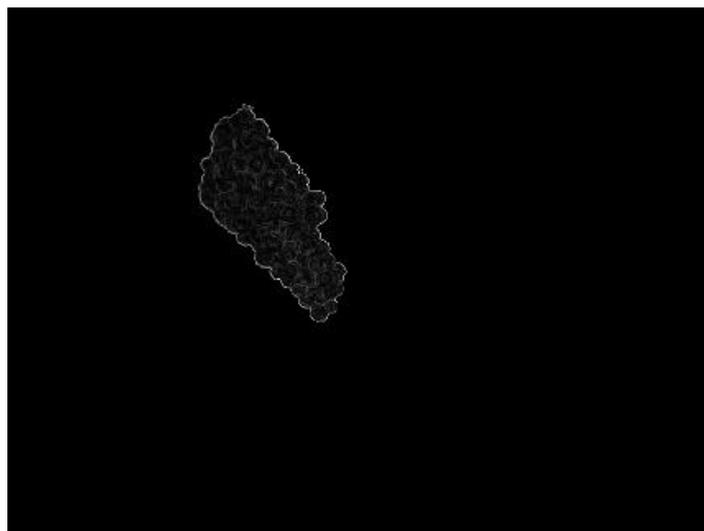


Figura 11. Magnitud del gradiente de I , enmascarada con la máscara del racimo.

Finalmente, se binariza la imagen utilizando un umbral. El valor del umbral se ha definido como la media de la magnitud del gradiente, de aquellos píxeles que lo tengan mayor a cero.

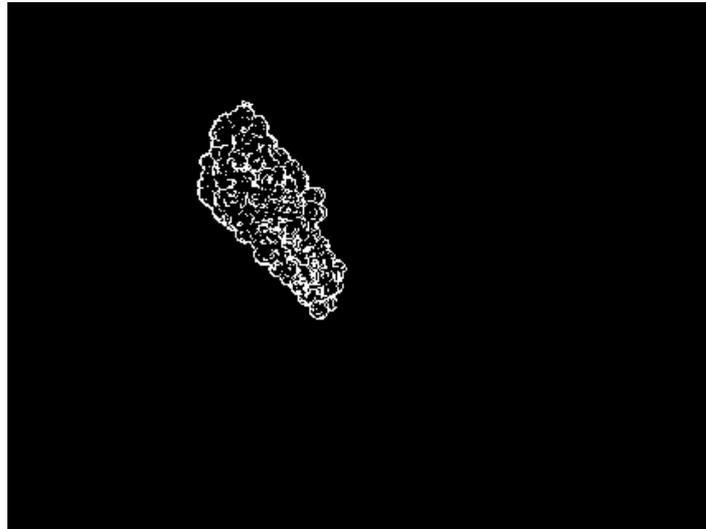


Figura 12. Resultado de la detección de bordes.

2.1.4. Estimación de círculos

La estimación de círculos se obtiene aplicando el Algoritmo Hough, el cual consiste en determinar los tres parámetros que definen a los círculos: radio (R), posición de centro horizontal (X) y vertical (Y). El algoritmo se divide en dos fases: acumulación y detección de picos, a las que se añade una tercera de filtrado.

Acumulación

Se define un espacio de parámetros (matriz) de 3 dimensiones, Figura 13, una dimensión por cada parámetro, es decir, R , X e Y . La matriz contiene todas las posibles combinaciones de R , X e Y que pueden tener los círculos a detectar, cada celda corresponde a un círculo de parámetros X_i , Y_j y R_k . Se recorren todas las celdas, inicializadas a cero, y se va acumulando (incrementando en 1) su valor por cada píxel de borde que está en el círculo que define los parámetros la celda. En la Figura 14 se muestra el resultado de la acumulación, pero para un solo valor de radio, el resultado final sería una combinación de imágenes como la mostrada, una para cada radio.

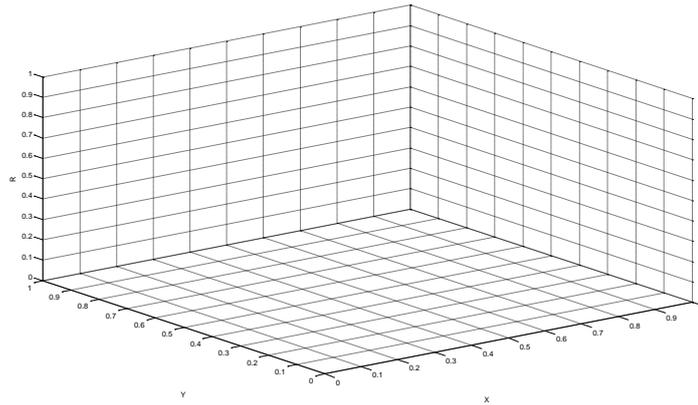


Figura 13. Matriz de parámetros de dimensiones (X, Y, R).



Figura 14. Matriz de acumulación Hough, para un radio específico.

Detección de picos

Cada uno de los puntos blancos que se observan en la figura anterior corresponde a un círculo centrado en esa posición y de un radio dado. Cuanto más alto es el valor (más blanco) del punto, más probabilidad tiene de ser realmente un círculo.

Para acotar el problema, se toman los 50 valores más altos (picos), éstos están ordenados de forma descendente y cumplen con la restricción geométrica de no estar a una distancia inferior a dos veces el radio del círculo respecto de los picos de mayor valor.

En un principio se pensó en detectar círculos en todo el rango de radios, que depende de cada clase de racimo, pero como se muestra en la Figura 15, hay la tendencia a detectar los círculos de radio menor, lo que es especialmente crítico si el tamaño predominante de las bayas es el mayor. Por ésta razón, se ha decidido separar la detección por radios, es decir, se buscan círculos de un solo radio cada vez, consiguiendo mantener los círculos potenciales

(posibles bayas), aunque se incremente el número total de detecciones y por consiguiente el número de falsas detecciones. Se vuelve necesario incluir una fase de filtrado de círculos.



Figura 15. Detección de círculos con rango completo de radios.

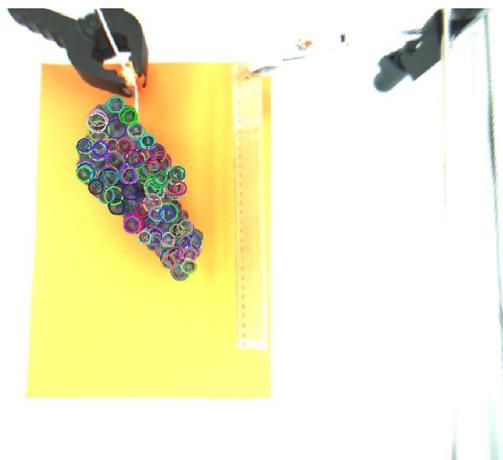


Figura 16. Detección círculos, combinando el resultado para cada radio del rango y filtrando con la orientación del gradiente

Filtrado

El filtrado se vuelve necesario para reducir el número de falsas detecciones, causadas por combinación de detecciones para distintos valores de radio. Para ello, se utiliza la otra parte de información que proporciona el gradiente de la imagen, la orientación (θ) que se calcula con (3).

La orientación⁶ en el contorno del círculo detectado (Figura 17.a) se compara con un patrón (Figura 17.b). La comparación establece un índice⁷, que mide lo parecida que es la

⁶ Se calcula el valor medio dentro de una máscara de 3x3, centrada en el píxel en cuestión.

orientación del círculo detectado respecto del patrón, si este índice no supera un umbral (establecido luego de hacer varias pruebas), entonces el círculo es admitido, sino se descarta, en la Figura 18 se muestra los círculos que son descartados y los que no.

Los círculos que se obtienen finalmente se muestran en la Figura 19. Círculos detectados con el Algoritmo Hough + Filtrado., el número sigue siendo alto, pero en gran medida son representativos de las bayas a detectar.

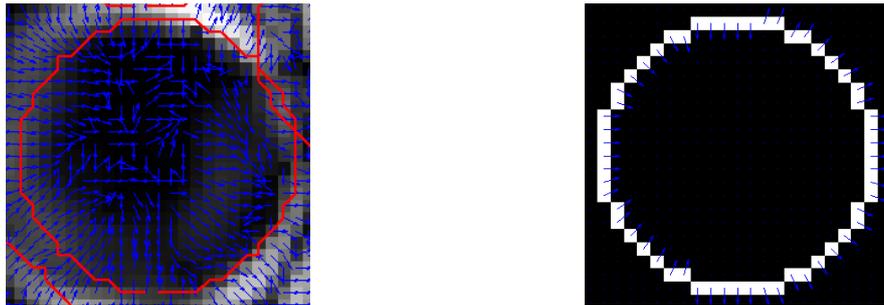


Figura 17. Orientación del gradiente. a. Círculo detectado. b. Patrón de comparación.

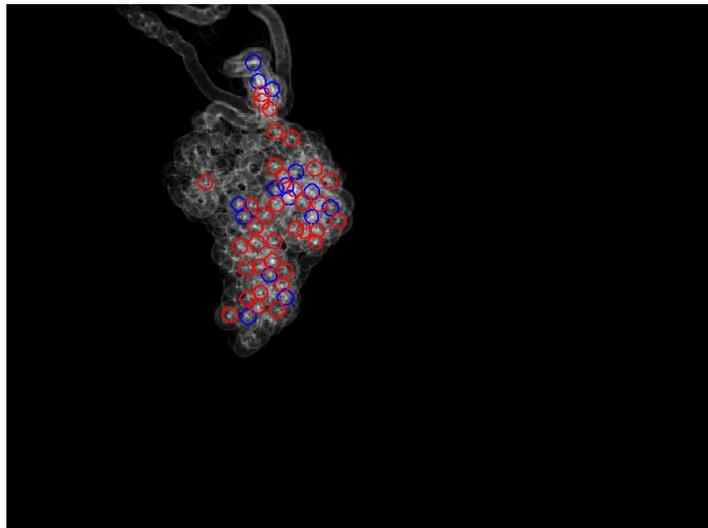


Figura 18. Círculos detectados para un radio dado. Pasan el filtro (rojos). Descartados (azul).

⁷ Cuanto menor sea el valor, más parecido hay con el patrón.



Figura 19. Círculos detectados con el Algoritmo Hough + Filtrado.

2.2. Búsqueda de correspondencias

La búsqueda de correspondencias es la tarea más compleja de todo el proceso de reconstrucción del racimo de uvas, esto se debe a la dificultad que conlleva el encontrar características que se manifiesten de manera adecuada en ambas imágenes del par estéreo, que lleven a la conclusión de que dos pares de círculos detectados corresponden a la misma baya.

En los objetos con curvatura, como es el caso (cuasi-esferas), la luz reflejada puede provocar que se modifique la apreciación de su proyección en una imagen de forma diferente que en su par estéreo, es lo que se conoce como efecto de la componente especular. En la Figura 20 se observa cómo afecta la luz, se forman unos bordes claros que forman un cuadrilátero, a las bayas. El reflejo de la luz aparece aproximadamente en el centro de las proyecciones de las bayas en el par estéreo de imágenes, y por tanto, sería un error intentar tomar éste efecto como característica para intentar el emparejamiento, ya que aunque el reflejo se localice en el centro, se produce en un área diferente de la baya.



Figura 20. Efecto de la componente especular. a. Izquierda. b. Derecha.

Otro de los problemas es la profundidad del racimo, ya que, cuanto mayor es la diferencia de profundidad entre dos bayas geoméricamente vecinas, mayor es la posibilidad de que se produzca un ocultamiento visual, esto sucede sobre todo con bayas localizadas en la zona central del racimo. La Figura 21 muestra como un par de bayas perfectamente visibles en la imagen izquierda, una de ella es ocultada casi totalmente por la otra en la imagen derecha.

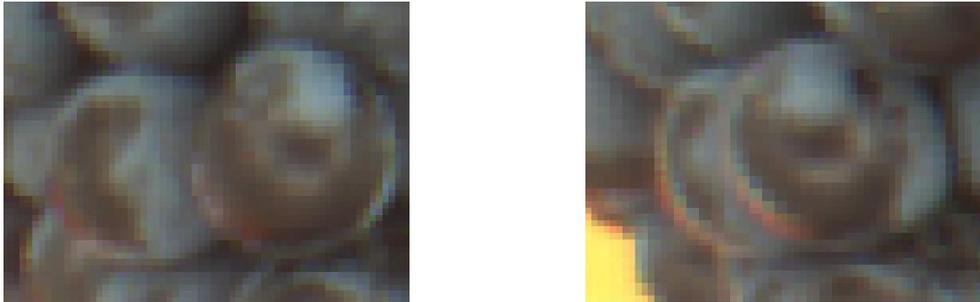


Figura 21. Ocultamiento visual debido a la diferencia de profundidad de bayas vecinas.

Algo similar ocurre en los extremos del racimo, pero esto se debe principalmente por el punto de vista desde el que se ha capturado cada imagen, Figura 22, de forma que, proyecciones de bayas ubicadas en el extremo izquierda de la imagen izquierda son casi siempre ocultadas en la imagen derecha, al igual que las proyecciones en el extremo derecho de la imagen derecha se ocultan en la imagen izquierda.

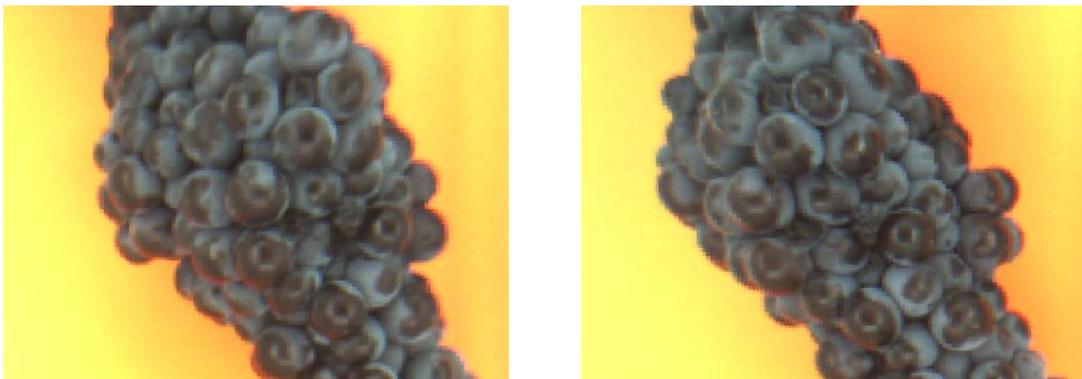


Figura 22. Ocultamientos de proyecciones de baya localizadas en los extremos del racimo.

La profundidad está directamente relacionada con la disparidad (d) (24) que hay entre el par de imágenes, siendo b la línea base de la cámara y f su distancia focal, que para ésta cámara es la misma tanto en horizontal como vertical, es decir, $f = f_x = f_y$.

$$Z = \frac{b \cdot f}{d} \quad (5)$$

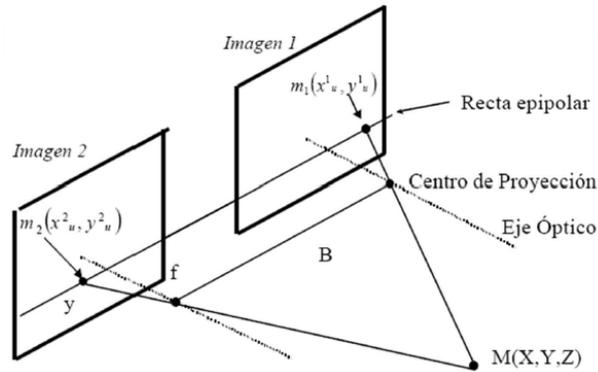


Figura 23. Parámetros que definen la cámara.

La disparidad se calcula como la diferencia de la posición horizontal entre un píxel de la imagen izquierda (X_{izq}) y su homólogo en la imagen derecha (X_{der}). A mayor disparidad, menor profundidad de la baya. En la Figura 24 se han tomado dos puntos homólogos en bayas de poca profundidad (más cercanas a la cámara), con una disparidad entre 239 y 249 píxeles. Por tanto, bayas vecinas deberían tener una disparidad similar, caso contrario, es muy probable que se produzcan ocultamientos, como ocurre en los casos mencionados anteriormente. Adicionalmente, se puede observar como a mayor disparidad, mayor es la diferencia en cuanto a color entre un par de proyecciones de baya homólogos, lo cual afecta a otras características como el gradiente.

$$d = x_{izq} - x_{der} \quad (6)$$

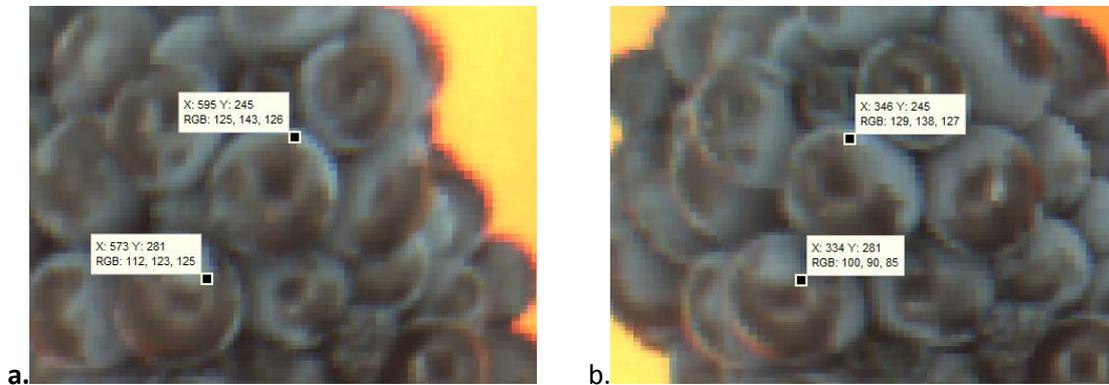


Figura 24. Disparidad entre puntos homólogos del racimo. a. Izquierda. b. Derecha.

Para resolver el problema de correspondencias se consideran tres aspectos que pueden ayudar a caracterizar las proyecciones homólogas de bayas:

- Geometría.
- Disparidad.
- Gradiente.

Geometría

Teniendo en cuenta que la detección se está haciendo para un valor de radio cada vez, el tamaño de la baya queda ya determinado, queda por comparar la posición del centro del círculo en dirección vertical (en horizontal no hay ninguna relación), que se hace considerando la restricción epipolar ($y_{izq} = y_{der}$).

Por lo tanto, dado un círculo en una de las imágenes, se consideran posibles parejas aquellos círculos de la imagen dual cuya posición en dirección vertical este dentro de un rango, establecido en 3 píxeles (7), si no se encuentra ningún círculo candidato puede deberse a que sencillamente no hay círculo dual (ocultamiento) o que no se ha detectado, lo cual puede deberse a que los bordes no están bien definidos (poca variación de color⁸).

$$\forall Y_{1i} \text{ es candidato de } Y_{2j} \text{ si } |Y_{1i} - Y_{2j}| \leq 3 \quad (7)$$



a.

b.

Figura 25. Candidatos que cumplen la restricción epipolar.

Disparidad

El parámetro del círculo que aún no se ha utilizado es la posición horizontal del centro, ésta se va a considerar por medio de la disparidad, de forma que, la banda de búsqueda que antes atravesaba todo el ancho del racimo, ahora estará, a priori, más acotado.

⁸ Para ser exactos, poca variación en la intensidad de la imagen en escala de grises.

Para establecer los límites de acotamiento, se van a considerar el valor de disparidad medio y el máximo. El valor medio se estima como la disparidad en la punta de la pinza, Figura 26, $552 - 370 = 182$ píxeles. De forma similar se hace con la disparidad máxima, Figura 24, $595 - 346 = 249$. Hay una diferencia de 67 píxeles entre la disparidad media y la máxima, por lo que el rango de variación se establece en 182 ± 67 píxeles.

$$rango = [115, 249]$$



Figura 26. Disparidad media.

Sin embargo, el recorrido del rango (134 píxeles) resulta bastante holgado, por lo que realmente será de utilidad cuando los racimos sean de una anchura mucho mayor, en caso contrario, el efecto es mínimo, como se puede apreciar en la Figura 27, donde, el rango en el que se puede localizar la posición horizontal del centro va de 338 a 476 píxeles, para un círculo con posición horizontal ubicado en 589 píxeles (Figura 26a), el resultado sería exactamente el mismo que al aplicar solo la restricción geométrica.

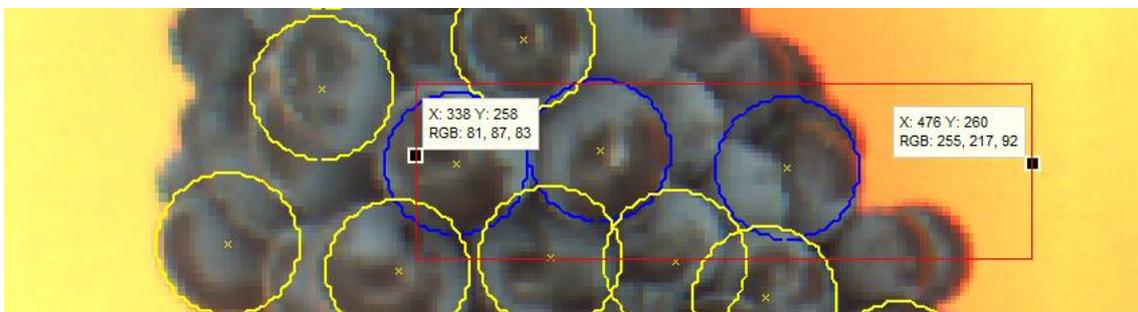


Figura 27. Definición del rango de variación de la disparidad

Gradiente

Como se ha visto hasta ahora, el gradiente está compuesto por magnitud y orientación. La magnitud se utiliza en fases de detección de bordes, mientras que la orientación en fases de comparación. A diferencia de como ocurre en la fase de filtrado de círculos detectados, donde se compara la orientación del gradiente en el borde del círculo con un patrón ideal del mismo radio (Figura 17), en esta fase se compara, además, la orientación en los píxeles interiores del círculo.

En la Figura 28 se representa un ejemplo de cómo un círculo en la imagen izquierda (a) es comparado con dos de sus posibles candidatos en la imagen derecha (b y c), donde, se observa cómo la orientación del gradiente entre a y b es más parecida que entre a y c. La cuantificación de ésta semejanza se hace calculando la distancia euclídea píxel a píxel.

$$d_{ij} = \sqrt{(g_{xi} - g_{xj})^2 + (g_{yi} - g_{yj})^2} \quad (8)$$

Previamente, para evitar tergiversaciones en el cálculo de la distancia euclídea, debido a que la orientación puede tener tanto valores positivos como negativos pero representar el mismo ángulo ($45^\circ \equiv -135^\circ$) y por tanto, incrementar de manera errónea el índice de comparación; es por ello que, se hace una estandarización de la orientación, de forma que el rango de variación es $[-90^\circ, 90^\circ]$ y además si una de las componentes del gradiente es significativamente dominante, entonces se desprecia la componente más pequeña.

Para proporcionar un poco más de robustez a éste índice de comparación, en lugar de comparar directamente la orientación, se compara la orientación media en la vecindad de cada píxel (máscara de 3x3).

El índice de comparación por orientación se define como el valor medio de todas las distancias d_{ij} . Por lo tanto, cuanto menor sea dicho índice, más probable es que se produzca el emparejamiento.

$$\forall \theta_i = \tan^{-1} \left(\frac{g_{yi}}{g_{xi}} \right) \mid g_{yi} \geq 4g_{xi} \rightarrow \theta_i = \text{sign}(g_{yi}) \cdot 90^\circ$$
$$\text{o } g_{xi} \geq 4 \cdot g_{yi} \rightarrow \theta_i = 0^\circ \quad (9)$$

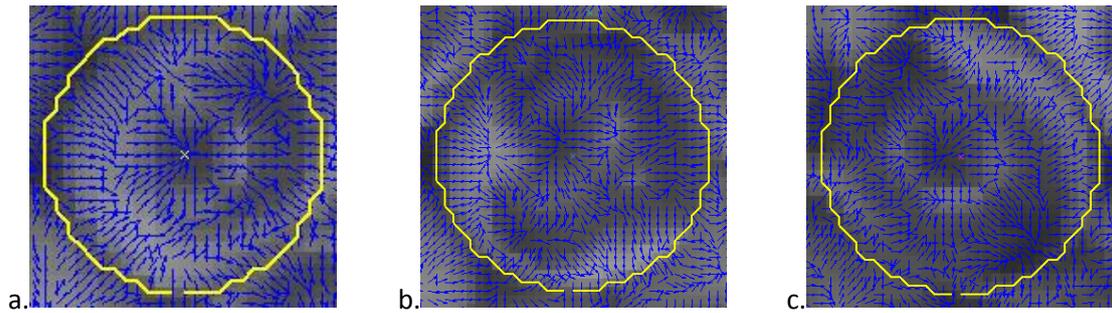


Figura 28. Comparación de la orientación del gradiente. a. Izquierda. b. Derecha candidato correcto. c. Derecha candidato erróneo.

Los tres criterios utilizados para la búsqueda de correspondencias, en teoría, deberían ser suficientes para determinar si se produce o no correspondencias entre círculos de la imagen izquierda y de la imagen derecha. En la práctica, se tiene un alto éxito en descartar emparejamientos poco probables, pero, cuando las posibles parejas para un círculo establecen índices muy semejantes, se vuelve temerario decidirse por una simplemente por tener un índice de comparación menor.

Por lo tanto, se ha decidido que si el primer y segundo mejores candidatos difieren poco en sus índices, se considerarán las dos parejas como válidas y más adelante cuando se haga la reconstrucción 3D se aplicarán otros criterios para dilucidar cuál es la pareja correcta.

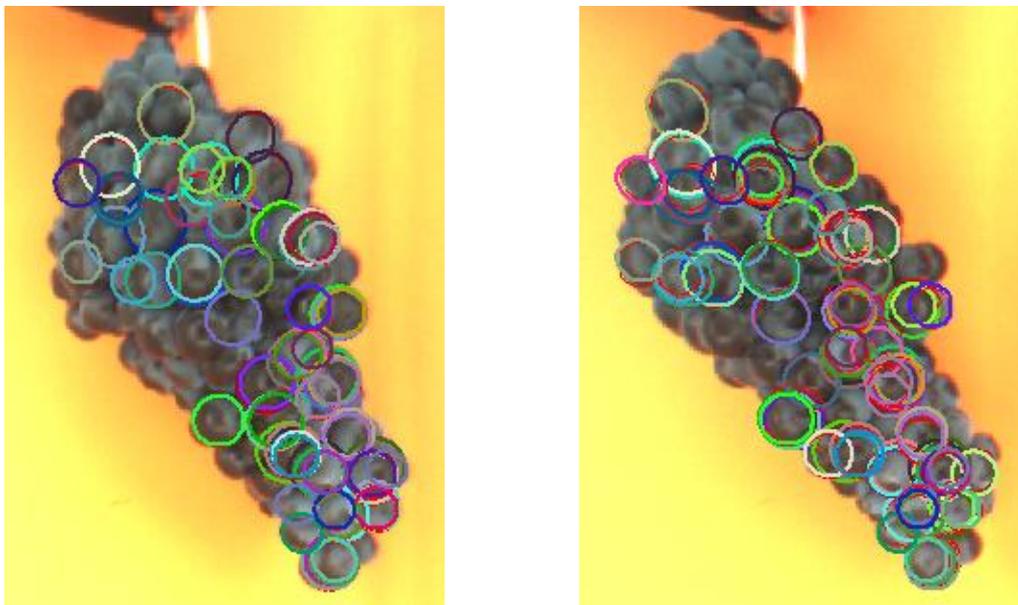


Figura 29. Resultado final de la búsqueda de correspondencias.

2.3. Reconstrucción 3D y estimación del tamaño de baya

2.3.1. Triangularización

Una vez que se tienen todas las posibles parejas de círculos (Figura 29), se procede a estimar las esferas cuyas proyecciones se aproximen a dichos círculos. Para ello se utilizan los parámetros extrínsecos e intrínsecos de la cámara.

Los parámetros extrínsecos son los que relacionan el sistema de referencia del mundo real y la cámara, describiendo la posición y orientación de la cámara en el mundo real. Lo componen las matrices de rotación (R) y traslación (T) tanto para la imagen izquierda como derecha, las cuales son determinadas durante la calibración de la cámara.

$$RT_{LEFT} = [R_{LEFT} \quad T_{LEFT}] \text{ y } RT_{RIGHT} = [R_{RIGHT} \quad T_{RIGHT}] \quad (10)$$

Donde, RT es la matriz de rotación-traslación de las imágenes izquierda y derecha, tomando para la cámara en cuestión los siguientes valores:

$$R_{LEFT} = R_{RIGHT} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad T_{LEFT} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad T_{RIGHT} = \begin{bmatrix} -0.1197 \\ 0 \\ 0 \end{bmatrix}$$

Este tipo de valores es típico de las cámaras estéreo con una configuración en paralelo, no hay rotación, una de las lentes está en el origen de coordenadas y la otra está desplazada horizontalmente. Los valores de desplazamiento están en metros.

Los parámetros intrínsecos son aquellos que definen la geometría interna y la óptica de la cámara. Son constantes, en tanto, no se modifiquen las características y posición relativa entre óptica y sensor de imagen. Si se cambia la lente, cambian las características intrínsecas de la cámara. Los parámetros se agrupan en forma de matriz M .

$$M = \begin{bmatrix} \alpha_x & 0 & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (11)$$

Donde,

$\alpha_x = \frac{f_x}{d_x}$, $\alpha_y = \frac{f_y}{d_y}$ distancia focal horizontal y vertical en píxeles.

f_x, f_y distancia focal en mm, como en ésta cámara $f_x = f_y$, se toma como f .

d_x, d_y , tamaño del píxel en x e y, viene dado en mm.

x_0, y_0 , punto principal expresado en píxeles.

En particular para ésta cámara toma los valores:

$$M_{LEFT} = M_{RIGHT} = \begin{bmatrix} 433.1090 & 0 & 512.6909 \\ 0 & 433.1090 & 395.9055 \\ 0 & 0 & 1 \end{bmatrix}$$

Otro de los parámetros que permanece constante es la línea base B (Figura 23).

$$b = 0.1197$$

Con toda la información ya definida, se procede a estimar cada una de las bayas que generan los pares de círculos obtenidos en la búsqueda de correspondencias.

En primer lugar se calcula la disparidad que hay entre sus centros, utilizando (6), para luego estimar con (5) la posición Z en el mundo real del centro de la baya.

$$\lambda = \frac{Z}{f} \quad (12)$$

A continuación, se calcula el factor de escalado (λ), que relaciona la esfera 3D en mm con su proyección en imagen en píxeles, por medio de (12). Éste factor de escalado es único para las tres coordenadas del mundo real (X, Y, Z) y por tanto se cumple que:

$$X = \lambda \cdot x \quad Y = \lambda \cdot y \quad (13)$$

Donde x e y corresponden a las posiciones horizontal y vertical del centro del círculo detectado.

Por lo tanto, ya se tiene localizado, en coordenadas del mundo real, el centro de la baya, desde donde se estima el radio de la baya utilizando una de las proyecciones y la matriz de parámetros intrínsecos M .

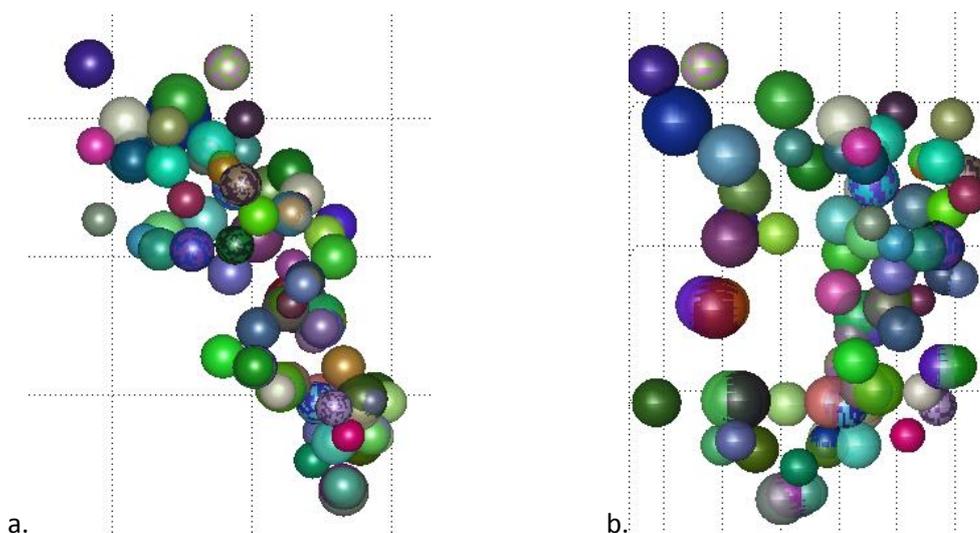


Figura 30. Reconstrucción 3D de todas las parejas de círculos detectados. a. Alzado. b. Perfil

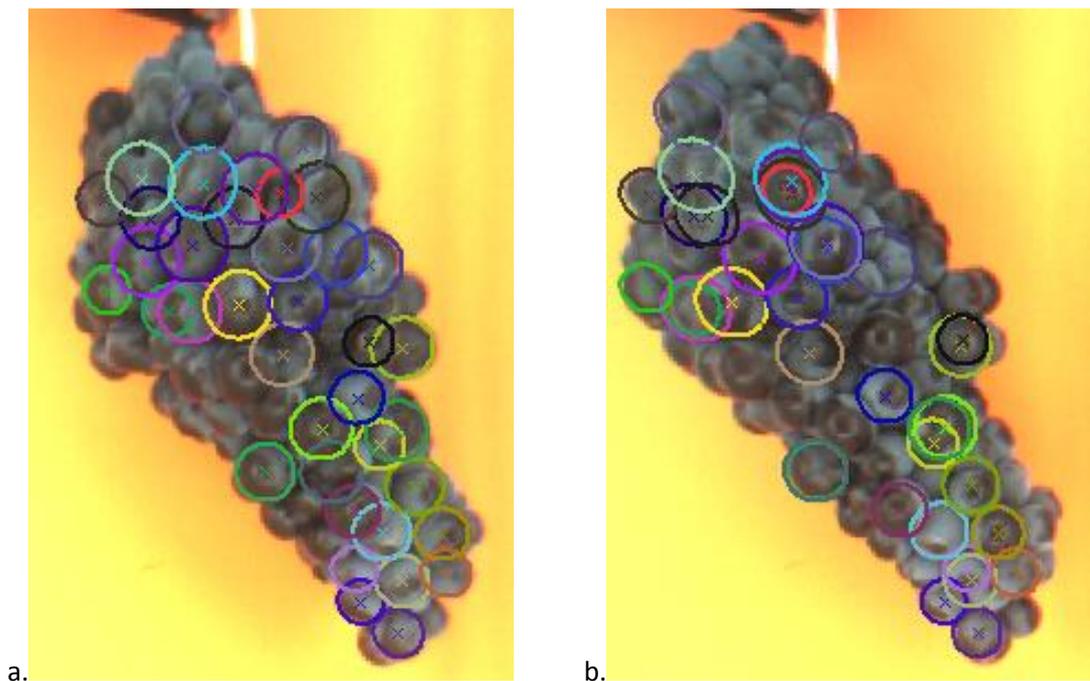
Esta secuencia de operaciones se repite para todos los pares de círculos detectados, obteniéndose el resultado de la Figura 30, cuyas proyecciones se muestran en la Figura 29. El número de esferas es elevado, y claramente hay esferas que o están solapadas o posicionadas en sitios anómalos (aislados, muy alejados u ocultados por esferas de menor profundidad), de ahí que sea necesaria una etapa posterior de refinamiento.

2.3.2. Refinamiento

Con el fin de reducir el número de esferas que no pertenecen al racimo, se va a emplear comparación por color. Para ello, se van a agrupar las proyecciones de las esferas según la distancia entre sus centros.

El proceso de eliminación de esferas mal detectadas se separa en dos fases. Primero se buscan grupos en la imagen izquierda y se comparan con sus respectivas parejas, la pareja que tenga mejor índice (valor más bajo) sobrevive, es decir, de cada uno de los grupos encontrados solo una de las esferas permanece. Puede darse el caso de que las esferas que quedan tengan distinta proyección en la imagen izquierda, pero muy similar en la imagen derecha (Figura 31a y b). Ésta es la razón de ser de la segunda etapa, donde, las proyecciones en la imagen derecha de las esferas restantes se agrupan y se comparan con sus parejas y nuevamente sobrevivirán las parejas de cada grupo que tengan mejor índice de color, Figura 32.

Adicionalmente, al observarse que en el racimo reconstruido había esferas localizadas en profundidades mucho más elevadas que la profundidad media, se ha incluido un filtro por disparidad (relacionada directamente con la profundidad), que permite eliminar las esferas más desplazadas en la dirección Z del sistema de coordenadas del mundo real.



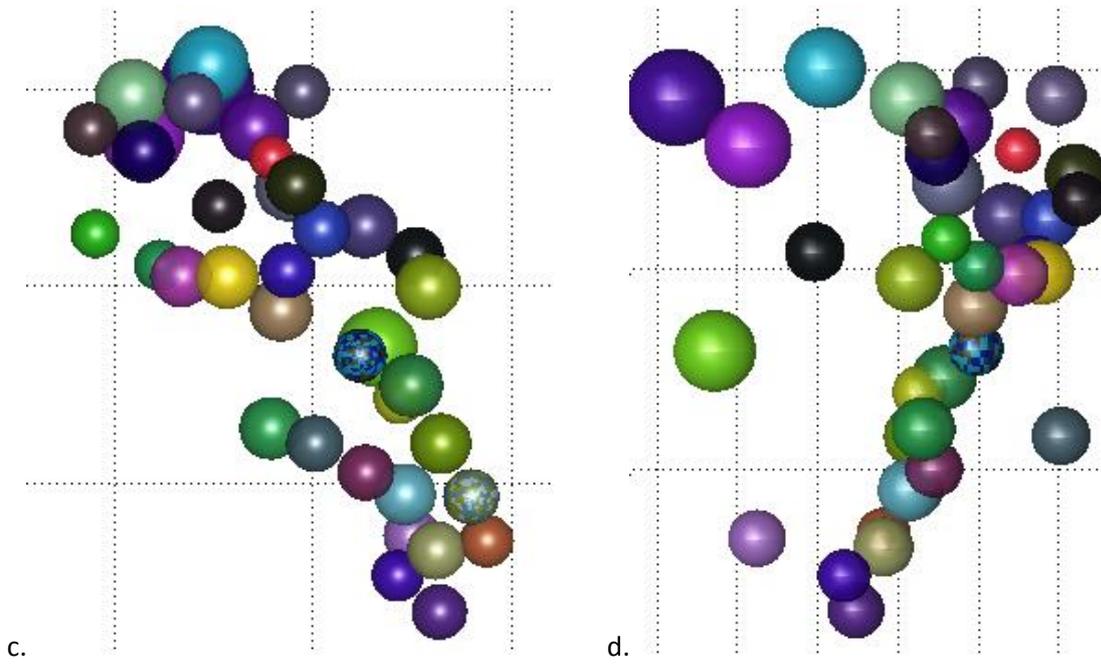


Figura 31. Primera etapa del refinamiento. a. Imagen izquierda. b. Imagen derecha. c. Alzado. d. Perfil



a.



b.

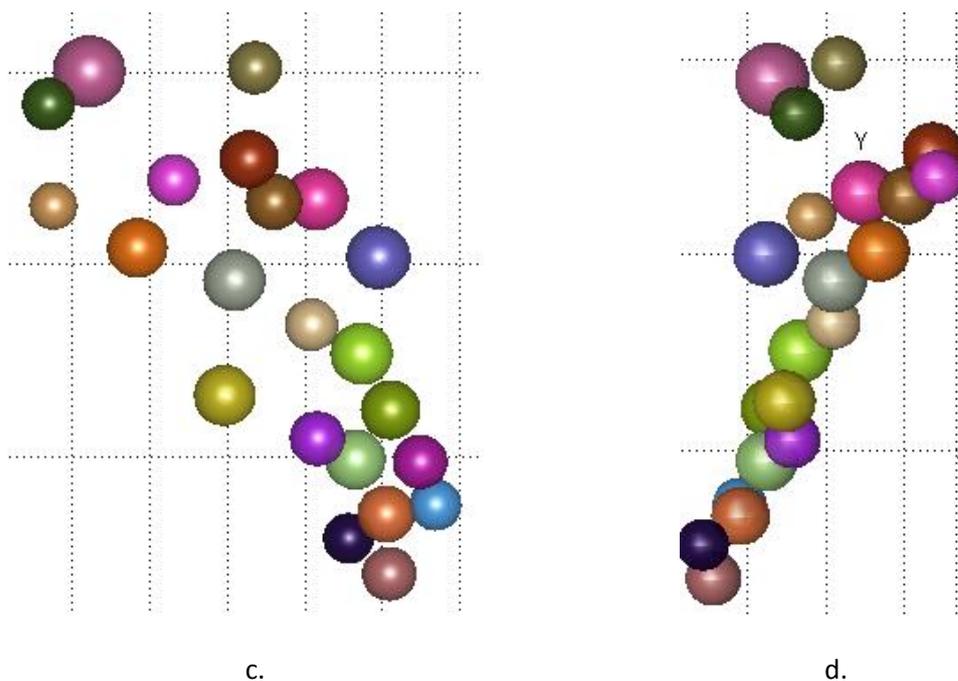


Figura 32. Segunda etapa del refinamiento. a. Imagen izquierda. b. Imagen derecha. c. Alzado. d. Perfil

Con las esferas finales se estima el radio medio de las bayas, utilizando tanto el valor medio como la mediana, de forma que sea lo más robusto posible.

$$R_m = \text{mean}(\text{mean}(R), \text{median}(R)) \quad (14)$$

2.4. Resultados

Todo el proceso descrito anteriormente se prueba sobre un conjunto de 10 racimos para cada una de las clases de uva de las que se dispone. En la Tabla 1 se muestran el número de bayas detectadas así como el radio estimado para cada uno de los racimos. En la Tabla 2 se muestra los errores de estimación tanto en la posición espacial de las bayas (posición del centro), así como el error del tamaño (radio) de la baya.

N	Bobal		Cabernet Franc		Cinsaut	
	N. círculos	R (mm)	N. círculos	R (mm)	N. círculos	R (mm)
1	21	7,267184	31	4,851459	17	7,883604
2	22	7,302066	33	5,23579	19	8,073203
3	22	7,328593	29	4,560259	18	8,237934
4	22	6,846741	21	4,678005	18	8,016934
5	28	7,043357	23	5,195595	15	7,833099
6	25	7,517507	24	4,463141	16	8,250184
7	18	7,088196	25	4,881044	13	8,172423

8	25	7,399231	23	5,062052	15	7,935907
9	33	7,152109	27	4,837343	7	7,725239
10	26	6,918876	28	4,741626	13	8,272755
Miguel de Arco			Monastrell		Moravia Agria	
N	N. círculos	R (mm)	N. círculos	R(mm)	N. círculos	R (mm)
1	26	6,377355	20	6,587849	26	5,856181
2	18	6,463224	16	6,798555	24	6,200492
3	21	6,153747	16	7,098661	21	5,944544
4	24	6,431221	18	7,058271	23	6,258329
5	19	6,414948	11	6,652301	14	5,850278
6	17	6,642939	18	6,313009	19	5,544226
7	15	6,784061	15	6,581393	20	5,879241
8	18	6,481712	21	6,792077	18	6,039288
9	18	6,835205	14	6,78843	24	5,768884
10	12	6,912981	14	7,043192	21	5,790724
Moscatel Morisco			Mouraton		Naparo	
N	N. círculos	R (mm)	N. círculos	R (mm)	N. círculos	R (mm)
1	21	6,574634	16	6,637458	26	7,41118
2	19	6,512063	17	6,877	30	7,058539
3	17	6,35399	19	6,774112	26	6,927244
4	22	6,353276	14	6,624654	30	7,171539
5	12	6,460811	17	6,455962	31	7,181354
6	22	6,37705	19	6,450688	25	7,0881
7	23	6,103749	18	6,534797	23	7,088484
8	10	6,717749	17	6,843275	28	7,185918
9	25	5,882489	23	6,78932	21	6,899765
10	29	6,032491	18	6,540796	36	6,609298
Planta Mula			Ruby_Seedless			
N	N. círculos	R (mm)	N. círculos	R (mm)		
1	15	7,047106	27	6,519684		
2	25	6,723556	21	6,667146		
3	20	6,747759	20	6,990525		
4	21	6,958295	23	6,581982		
5	20	6,360139	21	7,02828		
6	23	6,354377	24	6,90142		
7	28	6,534957	21	6,63472		
8	26	6,505724	18	6,924635		
9	25	7,129452	15	6,697572		
10	36	6,735723	22	6,811469		

Tabla 1. Resultados experimentales. Detección de círculos en 10 racimos para cada una de las variedades de uva disponibles. Número de círculos detectados y radio estimado.

	Bobal	Cabernet Franc	Cinsaut	Miguel Arco	Monastrell
error posición (mm)	2,29	2,19	2,16	2,64	2,97
error radio (mm)	0,54	0,56	0,51	0,53	0,44

	Moravia Agria	Mouraton	Naparo	Planta Mula	RubySeedless
error posición (mm)	2,64	3,43	2,57	2,06	2,78
error radio (mm)	0,68	0,64	0,61	0,66	0,76

Tabla 2. Resultados experimentales. Comparación de los modelos 3D obtenidos con los modelos 3D patrón de los racimos disponibles en la detección de círculos.

3. Elipses

La detección de elipses es una variante más compleja y completa de la detección de círculos. Las elipses están definidas por 5 parámetros: coordenadas X e Y del centro, eje (semi-eje) mayor, eje (semi-eje) menor y ángulo de orientación.

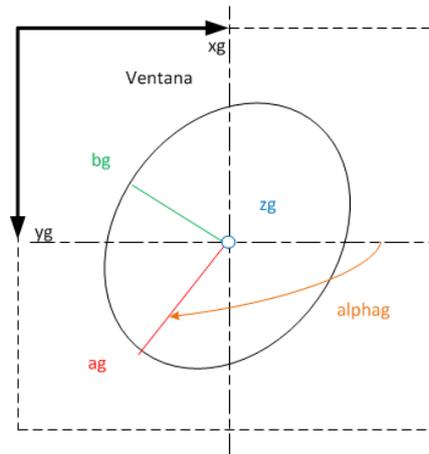


Figura 33. Parámetros que definen una elipse.

De entre la literatura revisada hay dos enfoques, interesantes, propuestos para la resolución del problema de la detección de elipses. Por un lado está el enfoque basado en el análisis de curvatura [7] [8], en donde se determina la curvatura de los bordes y se los agrupa según su afinidad para formar una elipse, la cual luego se estima. Por otro lado está el enfoque basado en modelos [6], que plantea la resolución del problema utilizando los bordes que se acomodan a un cierto patrón previamente establecido. En los siguientes apartados se describen ambos enfoques de detección.

3.1. Algoritmo de detección basado en la curvatura

El algoritmo de detección basado en la curvatura (algoritmo de curvaturas o algoritmo prasad), presentado en el estado del arte, permite la detección de elipses a partir de bordes basándose principalmente en el cálculo de su curvatura. Los resultados obtenidos al implementarlo de forma directa, tal cual lo propone su autor –Prasad–, sobre las imágenes que se utilizan en este trabajo no son muy satisfactorios, pues, tanto el número de elipses detectadas como la precisión con que se detectan son bajos, como se había mostrado en la Figura 2. Por lo tanto, se propone introducir algunas modificaciones para salvar los inconvenientes que se han encontrado durante su implementación directa, con el fin de mejorar los resultados iniciales.

La característica que define al algoritmo es el cómo se tratan los bordes. Se pretende estimar la curvatura que tienen los bordes, a partir de la cual, definiendo un área de búsqueda, se forman grupos afines. Estos grupos están formados por bordes que pueden, dependiendo del valor de curvatura que tengan, pertenecer a una misma elipse.

Los principales inconvenientes a corregir son el número de elipses detectadas, el número de falsas detecciones y el incremento puntual (o permanente) en el tiempo de cómputo. Los cuales se intentan solventar incluyendo restricciones a los bordes, pues son éstos sobre los que se centra todo el algoritmo.

3.1.1. Restricciones en los bordes

El algoritmo parte de una imagen binaria que representa los bordes detectados a partir de una imagen RGB, en éste caso se ha utilizado un detector de bordes Canny con un umbral de detección previamente establecido. Con un umbral de 0.1 el algoritmo llegó a bloquearse, mientras que con un umbral de 0.2 el algoritmo se ejecutó con normalidad obteniéndose el resultado de la Figura 2.

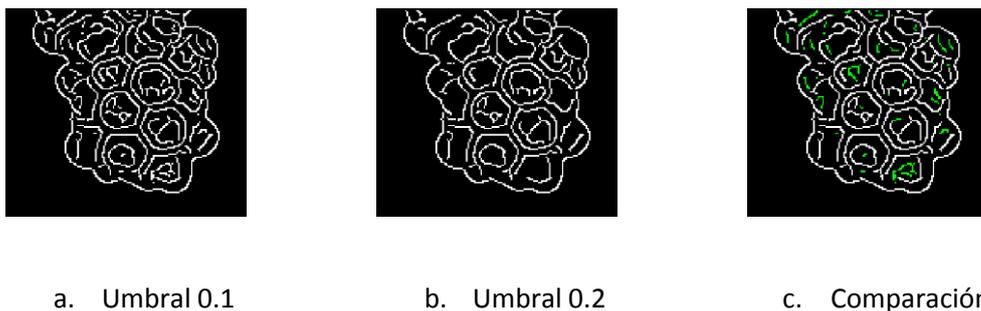


Figura 34 Detección de bordes utilizando Canny para distintos umbrales.

Los bordes obtenidos con el detector Canny aumentan cuando la imagen contiene muchos detalles y/o el umbral es muy bajo, en la Figura 34 se muestra como varía el número de bordes detectados con el cambio de umbral. Un incremento en el número de bordes implica que la probabilidad de detectar bordes sin curvatura se eleve.

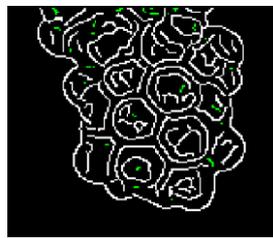
El incremento en el tiempo de cómputo es causado principalmente por los bordes sin curvatura, que en primer lugar consumen tiempo para ser clasificados como tales y en segundo lugar provocan que el área de búsqueda de afinidades (Figura 1.d) sea más grande, por lo que se incrementa el número de comparaciones entre los bordes al haber, generalmente, más bordes que comparar dentro del área de búsqueda. Además está el hecho de que cuanto mayor es el número de bordes, mayor es el tiempo de cómputo puesto que hay más bordes a los cuales definir su curvatura; por lo tanto, hay que llegar a un compromiso entre bordes detectados y elipses que se quisieran detectar fielmente.

Observando la Figura 34.c se aprecia los bordes adicionales introducidos por el uso de un umbral de detección bajo. A partir de ellos se puede interpretar cuales son las razones por las cuales el algoritmo ralentiza en exceso su tiempo de ejecución o incluso llegar a bloquearse.

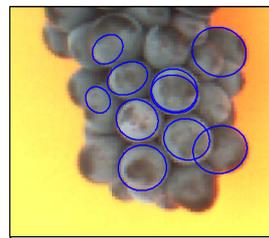
Se consideran a priori 3 razones principales, relacionadas entre sí:

- Bordes muy pequeños.
- Bordes con curvatura nula (líneas rectas).
- Número de bordes elevado.

Para tratar de solventar estos inconvenientes se plantea como solución filtrar los bordes más pequeños, estableciendo una longitud mínima de borde.



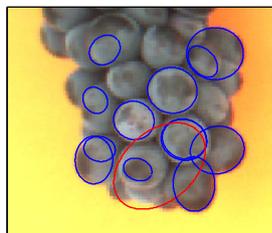
a. Bordes pequeños



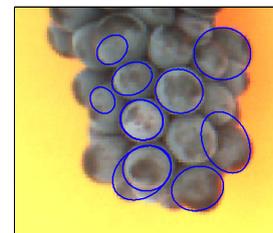
b. Elipses detectadas

Figura 35 Introducción de restricciones en el tamaño de bordes. Canny con umbral 0.1

La curvatura de los bordes se estima a partir del ángulo que forman puntos del borde seleccionados aleatoriamente. Esta aleatoriedad en ocasiones provoca que algunas elipses detectadas sean falsos positivos, y por contra, otras que deberían ser detectadas fácilmente no lo hacen. Con las modificaciones introducidas se corrige en una cierta medida ese problema.



a. Prasad con eliminación de bordes pequeños.



b. Prasad con restricción de tamaño máximo y mínimo.

Figura 36 Detección de elipses falsas como verdaderas.

Puede aparecer el problema de detectar elipses de tamaño desproporcionadamente alto, como la que se muestra en la Figura 36.a, lo cual se debe a la baja curvatura de los bordes, ya que solo las elipses de gran tamaño se adecuan a tal curvatura con un bajo error de interpolación.

Adicionalmente, se ha comprobado como la aleatoriedad, con la que la curvatura de los bordes es calculada, introduce como desventaja que la solución obtenida, manteniendo la misma configuración de parámetros, no es siempre la misma o al menos parecida, es decir, mantener el número de elipses detectadas aunque sus dimensiones no sean exactamente las mismas. Una solución se considera factible si las elipses principales se detectasen siempre. En la Figura 37 se muestra un ejemplo de solución no factible, ya que hay elipses relevantes que son detectadas unas veces y otras no.



Figura 37 Incertidumbre en la solución obtenida causada por la aleatoriedad en el cálculo de la curvatura.

Hasta el momento se ha tratado la imagen como un todo, o al menos una región de interés (ROI) grande, por lo que para de minimizar los efectos de la aleatoriedad en la solución, se modifica el concepto de ventana (window).

El algoritmo utiliza la ventana para definir el área de búsqueda, que es donde se encontrarían los bordes afines. La ventana tiene unas dimensiones fijas y se crean a partir de los bordes que se hayan detectado, siendo éste factor el que se modifica. Ahora, la ventana o máscara, se utiliza para recorrer la imagen completa (independientemente de los bordes), fijando un 'paso' de avance tanto en horizontal como en vertical, lo que permite detectar más de una elipse para una misma uva, reduciendo la posibilidad de que debido al no determinismo del algoritmo no se detecten elipses significativas.

Por lo tanto, el algoritmo se aplica sobre cada una de estas ventanas, detectando las correspondientes elipses, que posteriormente se unifican para formar la solución global. Sobre ésta solución se realiza un agrupamiento, según la distancia entre centros de las elipses. Se establece un límite de distancia entre elipses por debajo del cual formarían un grupo. En cada grupo se estima una elipse resultante, que formarán la solución final.

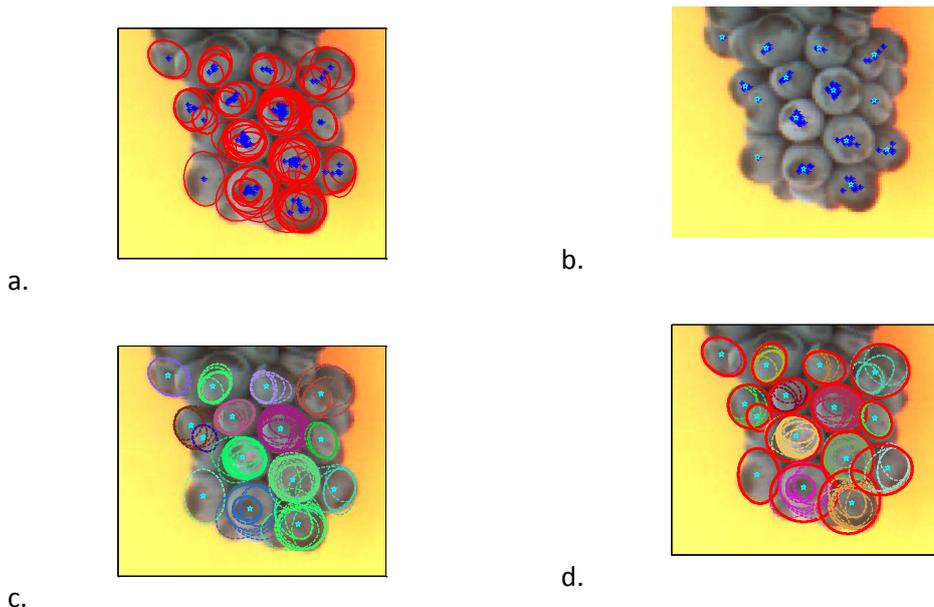


Figura 38. Algoritmo Prasad con ventanas y restricción del tamaño del borde. a. Elipses detectadas. b. Centros de las elipses detectadas y centro medio de las elipses agrupadas. c. Elipses agrupadas. d. Elipses resultantes de cada grupo.

La inclusión de las ventanas en el algoritmo causa que, efectivamente, la gran mayoría de uvas tengan su elipse que la represente (Figura 38.a), si bien, queda por optimizar tanto el agrupamiento (Figura 38.b y Figura 38.c), en esta primera aproximación se han agrupado las elipses cuyos centros estén a una determinada distancia; como la estimación de las elipses resultantes, calculada de forma lineal a partir de los puntos que definen las elipses en cada uno de los grupos, que como se observa en la Figura 38.d no representa a las uvas fielmente.

Sin embargo, se ha conseguido solventar los tres inconvenientes planteados, que es lo que en principio se pretendía. Éste es el alcance que tendría la detección de elipses con el algoritmo Prasad adaptado a las imágenes y circunstancias que se presentan en este trabajo.

3.1.2. Ventajas e inconvenientes

La mayor ventaja que presenta el algoritmo de curvaturas es la sencillez en la implementación. Se trata de un algoritmo bastante potente y preciso, sobre todo, cuando se dispone de bordes de una buena calidad. Es por ello que se plantea un procesado de imagen más específico para el tipo de imágenes que se están utilizando y lo que se quiere detectar en ellas.

A pesar de su sencillez, presenta el inconveniente de que al tratar de hallar la curvatura en bordes que son prácticamente rectos, a menudo, entra en un bucle de cálculo. Por lo tanto, además ya de por sí tener inherente un alto coste computacional, éste puede volverse tan elevado que puede llevar a descartar su utilización.

3.2. Algoritmo de detección basado en modelos.

El algoritmo de detección basado en modelos presentado o algoritmo pedestrians en [8] [7], donde el objeto de detección son personas (las siluetas de personas), es adaptado para tratar de resolver la detección de elipses en las imágenes de racimos de uvas con las que se está trabajando. Se realizan los cambios necesarios, ya sean modificaciones en la forma de implementarlo o suprimiendo pasos que tienen más sentido en detección de personas, más no para elipses.

3.2.1. Desarrollo

El punto de partida son las imágenes en escala de grises, obtenidas del mismo modo que se ha mencionado con anterioridad, sobre la cual se buscan las características de borde. Éstas características son la magnitud y orientación del gradiente, que se aplica directamente sobre las imágenes en escala de grises.

El concepto de ventana se mantiene, incluido sus dimensiones⁹. Lo que cambia, obviamente, es lo que se hace en cada ventana. En este algoritmo se pretende hacer una clasificación previa de la ventana, para determinar si puede contener o no una elipse.

La primera parte del algoritmo consiste en obtener los patrones que definen las bayas, por lo tanto hay que llevar a cabo una fase de entrenamiento. El entrenamiento se hace a partir de un conjunto de imágenes. Las imágenes son separadas en dos grupos: imágenes positivas e imágenes negativas, con las que se realizará el entrenamiento de forma independiente.

Las imágenes positivas son aquellas en las cuales se puede detectar una elipse, que deberá cumplir unos criterios de validez, mientras que las imágenes negativas son aquellas en las que no se encuentran elipses, que cumplan los criterios, o cumplan otros que las defina como negativas.

Las características que debe cumplir la baya detectable en una imagen para que sea considerada positiva son:

1. Hay solo una.
2. Estar localizada aproximadamente en el centro de la imagen (ventana).
3. Tener visible más de la mitad de su contorno.

Las imágenes de la Figura 39 son ejemplos de imágenes de entrenamiento que se han considerado como positivas.

En cuanto a las imágenes negativas, se ha considerado como criterio que haya bordes de baya en la zona central de la imagen, aunque lo ideal y más apropiado sería generar patrones

⁹ Se mantienen las dimensiones tanto de la ventana como del paso de recorrido para que las propuestas puedan ser comparadas objetivamente.

para distintos tipos de configuraciones negativas, pero esto haría más complejo y costoso la implementación del algoritmo. En la Figura 40 se muestran algunos ejemplos de imágenes que se han tomado como muestras negativas.

Una vez que se toman un número suficiente de imágenes de entrenamiento, se obtienen los patrones positivo y negativo de las imágenes, para todas las características definidas, que en este caso son la magnitud y orientación del gradiente. A partir de estos patrones se obtendrán los modelos tanto de magnitud como de orientación.

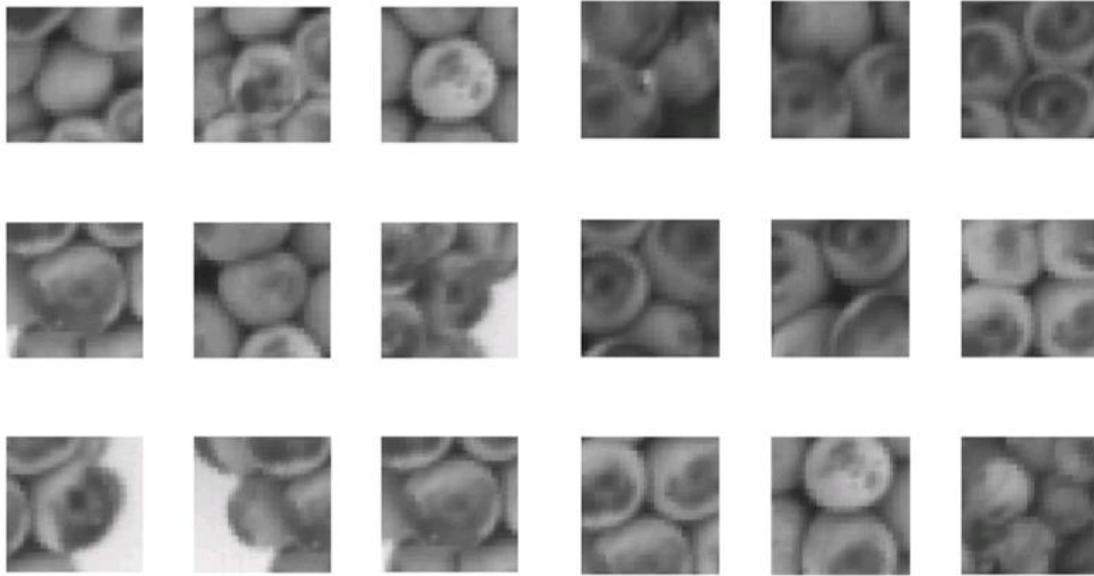


Figura 39. Ventanas de entrenamiento positivas

Figura 40. Ventanas de entrenamiento negativas

Modelo de magnitud

Previamente, se realiza una normalización de los gradientes obtenidos de cada ventana, para que las decisiones tomadas (definición de parámetros como umbrales) sean generalizados; por tanto, la magnitud varía entre 0 y 1. Debido a que la magnitud siempre va a ser positiva, se calcula (directamente) la media de todas las ventanas en cada una de las posiciones de los píxeles. El resultado es el modelo de probabilidad de magnitud, donde, cuanto mayor sea el valor en la posición del píxel, mayor será la probabilidad de que sea un borde la baya.

$$g_x = \frac{\Delta I_x}{\Delta \text{píxeles}_x}; g_y = \frac{\Delta I_y}{\Delta \text{píxeles}_y} \quad (15)$$

$$G = \sqrt{g_x^2 + g_y^2} \quad (16)$$

Los gradientes horizontal y vertical son obtenidos con (15), utilizando la función propia de MATLAB 'gradient'. La magnitud se calcula con (16). El modelo de probabilidades positivo (PP), que se aprecia en la Figura 41, muestra como la zona central tiene poca probabilidad (prácticamente nula) de albergar un borde de elipse. En cambio, el anillo centrado es la zona donde hay más posibilidad de que hayan bordes, mientras que la zona exterior tiene una menor aunque no nula probabilidad de contener bordes. Este comportamiento es el que, a priori, se esperaba para las imágenes positivas.

De igual forma se obtiene el modelo de probabilidades negativo (PN), Figura 42, donde se aprecia que en la zona central está la mayor probabilidad de tener bordes, que hagan de la imagen no válida para albergar una elipse que represente una baya.

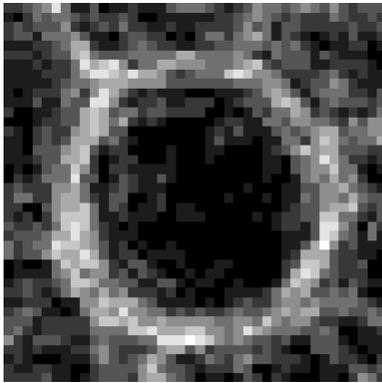


Figura 41. Modelo de probabilidades positivo (PP) de magnitud.

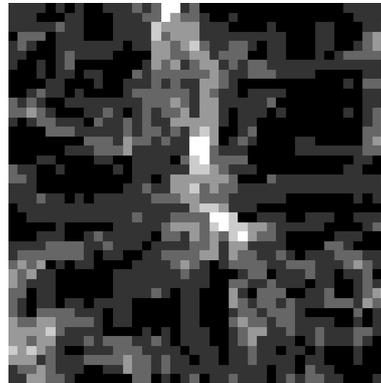


Figura 42. Modelo de probabilidades negativo (PN) de magnitud.

El siguiente paso es obtener el modelo de pesos, que definirá la importancia que tiene cada píxel. Para ello se calcula la desviación típica de la magnitud del gradiente en todos los píxeles, el inverso es el modelo de pesos.

La Figura 43 y Figura 44 muestran los modelos de peso positivo (WP) y negativo (WN) respectivamente. Se puede observar que se da más importancia a la zona donde hay menos probabilidad de localizar un píxel de borde válido, es decir, se prioriza el descarte de imágenes negativas sobre la aprobación de imágenes positivas.

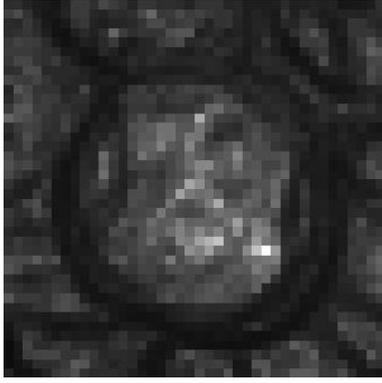


Figura 43. Modelo de pesos positivo (WP) de magnitud.

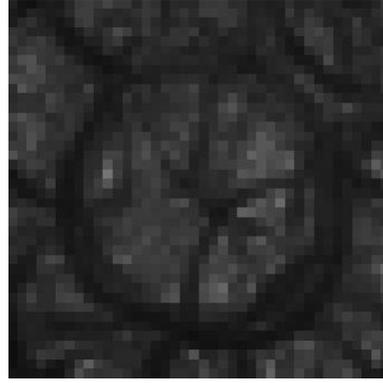


Figura 44. Modelo de pesos negativo (WN) de magnitud.

El modelo de magnitud (Figura 45) se obtiene, en primer lugar, multiplicando píxel a píxel los modelos de probabilidad y peso tanto de las imágenes positivas como de las negativas, y luego se resta el resultado negativo del positivo.

$$Modelo_M = PP \cdot WP - PN \cdot WN \quad (17)$$

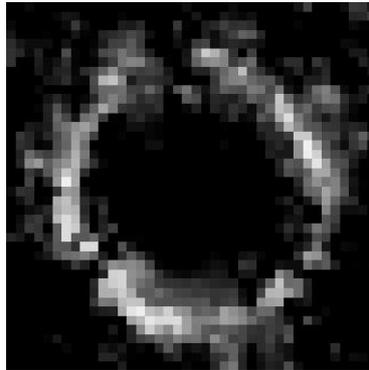


Figura 45. Modelo de magnitud

Finalmente, se define un umbral de clasificación, el cual en caso de ser superado cataloga la imagen como positiva y en caso contrario como negativa.

$$SW_{M_i} = \sum Modelo_M \cdot G_i \quad (18)$$

$$threshold_M = media(SW_{M_i}) \quad (19)$$

Para determinar el umbral, se calcula el peso de cada muestra i (SW_{M_i}) utilizando (18), donde, la multiplicación del modelo de magnitud ($Modelo_M$) y la magnitud del gradiente de la

muestra (G_i) es píxel a píxel y el sumatorio se hace sobre todos los éstos píxeles resultantes. La media de los pesos de las imágenes (19) define el umbral de clasificación.

Modelo de orientación

El modelo de orientación sigue la misma pauta que el modelo de magnitud. Sin embargo, hay unos aspectos a tener en cuenta y que lo hacen más enrevesado de implementar.

$$\theta = \tan^{-1} \left(\frac{g_y}{g_x} \right) \quad (20)$$

En primer lugar, el cálculo del valor medio de la orientación (20) no se puede hacer de forma directa ya que los valores que toma pueden ser positivos o negativos; por lo que, la media tendería a ser casi siempre nula, por lo tanto, la media dejaría de ser un parámetro representativo de la imagen. Para salvaguardar este inconveniente se opta por separar en 4 tipos¹⁰, según el valor de la orientación:

Tipo 0: $0 \leq \theta < 10$, $170 \leq \theta \leq 180$

Tipo 1: $10 \leq \theta < 80$

Tipo 2: $80 \leq \theta < 100$

Tipo 3: $100 \leq \theta < 170$

Se obtiene el modelo de probabilidades (PO) del mismo modo que ocurría en la magnitud, solo que ahora habrán 4 modelos, uno para cada caso. Figura 46.

El modelo de pesos de orientación (WO) se obtiene de igual forma que en magnitud, cuyo resultado se observa en la Figura 47, aunque como se aprecia, en este caso se da más importancia a la detección de imágenes positivas, como se deduce del anillo centrado que claramente puede ser distinguido.

¹⁰ Los valores que van de 180 a 360 se tratan como un espejo de los valores entre 0 y 180.

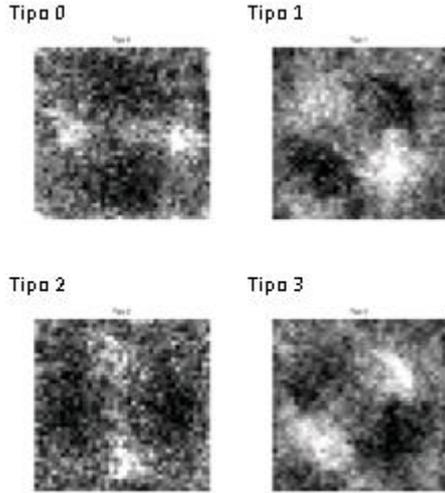


Figura 46. Modelos de probabilidad de orientación (PO)

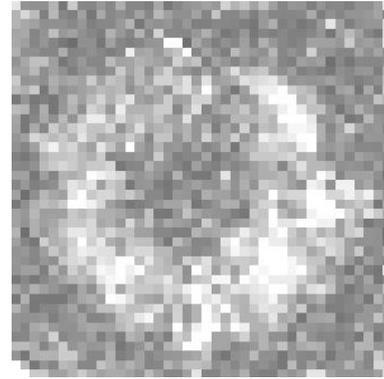


Figura 47. Modelo de pesos de orientación (WO)

Cabe mencionar que los modelos arriba representados corresponden al resultado final de combinar los modelos positivo y negativo de orientación, combinados de forma análoga al caso de magnitud mencionado anteriormente.

$$SW_{O_j} = \sum_{tipo=0}^3 (PO \cdot \theta_{tipo_i}) \cdot WO \quad (21)$$

$$threshold_o = media(SW_{O_j}) \quad (22)$$

El peso de la muestra se define mediante el sumatorio de la composición del modelo de probabilidad, el modelo de pesos y la orientación (clasificada por tipos), según (21), en donde:

- SW_{O_j} : Es el peso de la imagen j.
- θ_{tipo_i} : Vale 1 en los píxeles cuya orientación (θ) es del tipo i y 0 en los demás. Siendo $i \in \{0, 1, 2, 3\}$.

El umbral de caracterización por orientación queda definido por la media del peso de todas las muestras (22).

Implementación sobre el racimo

Una vez que se tiene establecido los modelos de magnitud y orientación así como sus umbrales de caracterización, además de la ventana tanto el tamaño como el paso de avance, se recorre toda la imagen (racimo), se calcula el peso de la ventana según magnitud (SW_M) y

orientación (SW_O), utilizando los respectivos modelos y empleando las ecuaciones (18) y (21). Los pesos se comparan con los umbrales¹¹ correspondientes, de forma que si se cumple que:

$$SW_M \geq threshold_M \text{ Y } SW_O \geq threshold_M$$

La ventana se considera positiva, y por tanto en ella hay una elipse que detectar. En caso contrario se considera negativa y se descarta.

El método de estimación de la elipse que se utiliza se detalla posteriormente en el algoritmo propuesto, aunque, aprovechando los modelos obtenidos se hace una ponderación previa, descartando los píxeles (válidos) menos fuertes de la ventana. Se escogen el 10% de píxeles totales que mayor valor tengan. En la Figura 48 se muestra una comparación entre los bordes totales y los ponderados que se utilizan en la estimación de la elipse, y claramente la reducción de píxeles de borde con los que tratar es significativa. En la Figura 49 se muestra la estimación de elipse obtenida y representada sobre la imagen (ventana) de grises inicial, y como se puede apreciar, es bastante fiel a la baya en cuestión.

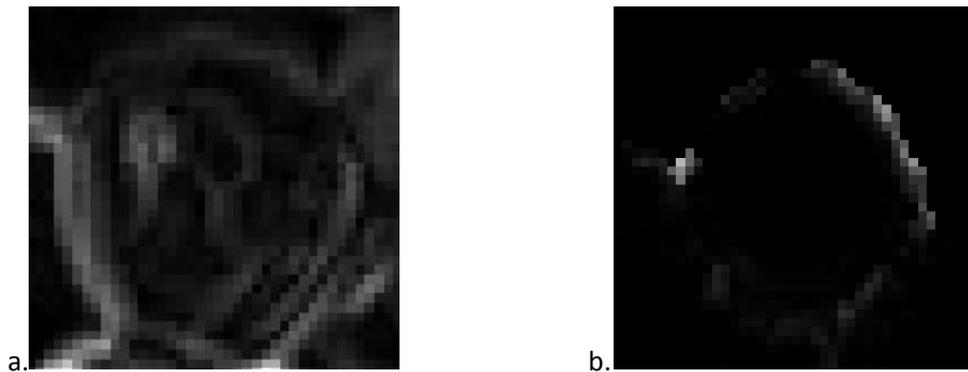


Figura 48. a. Magnitud del gradiente. b. Magnitud de gradiente validado con el modelo y ponderado.

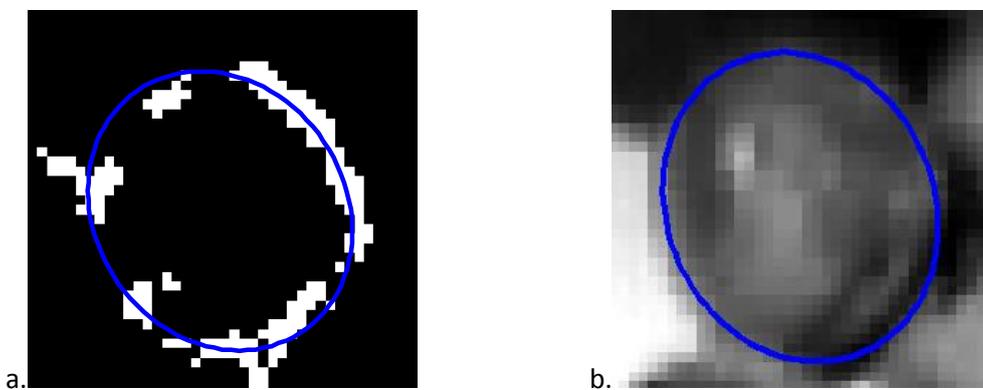


Figura 49. Estimación de la elipse, dibujado sobre a. Bordes ponderados binarizados y b. Imagen en escala de grises

¹¹ Se han reducido un 10% para asegurar que no se pierden ventanas válidas.

La implementación del algoritmo sobre un racimo entero arroja como resultado las elipses detectadas que se muestran en la Figura 50.a, donde de manera similar a como ocurría en la primera propuesta se parte de la idea de que las bayas son detectadas en más de una baya, y por tanto, se pueden agrupar según la distancia euclídea entre sus centros (Figura 50.b), de donde se estiman las elipses de cada grupo (Figura 50.c).

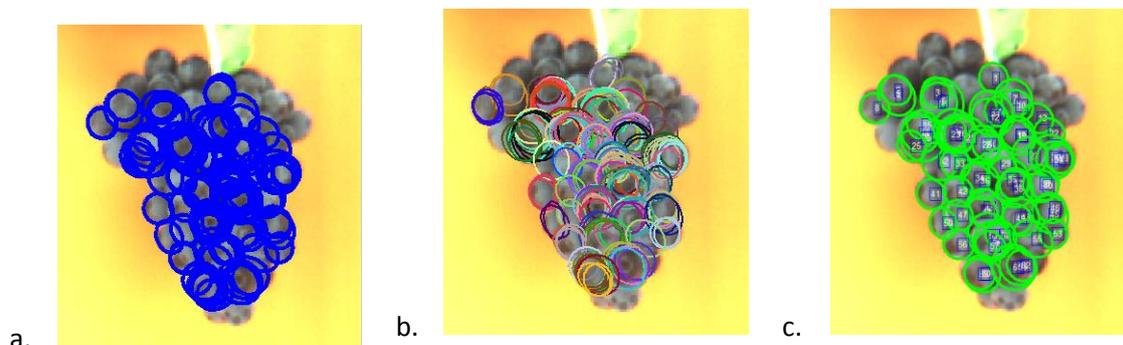


Figura 50. Implementación del Algoritmo Pedestrian sobre un racimo. a. Elipses detectadas. b. Agrupamiento. c. Estimación elipses de grupos.

3.2.2. Ventajas e inconvenientes

El algoritmo de detección basado en modelos tiene como ventaja que la cantidad de parámetros a definir, una vez que se obtiene el modelo, es pequeña. Por otro lado, el gran inconveniente que tiene es que si se quiere obtener unos resultados más o menos adecuados el proceso de entrenamiento para la consecución del modelo, que ya de por sí lleva mucho tiempo, es arduo.

La variabilidad de tamaño y aspecto de las bayas de uva no es muy elevado, por lo que obtener un único modelo para todos los casos es viable, siempre que se mantenga dicho ratio de variabilidad.

3.3. Algoritmo propuesto

A pesar de la facilidad de implementación del algoritmo Prasad, ya se vieron los resultados obtenidos, no demasiado halagüeños para las imágenes tan complejas con las que se trata (sombras, reflejos, poco contraste...), se propone un algoritmo que busque detectar las elipses de manera más precisa.

En esencia, del algoritmo Prasad se toman dos ideas: la ventana y la búsqueda de características. Además, el procesamiento de imagen es básico, se parte de una imagen RGB se convierte a escala de grises y se utiliza el detector de bordes Canny, por lo que es aquí donde se centrarán esfuerzos para mejorar la detección.

3.3.1. Procesado de imagen

La calidad de los bordes es fundamental a la hora de plantear la detección de elipses, por tanto, el procesamiento de imagen va encaminado a mejorar la calidad. En particular, las ideas que se plantean se hacen considerando las características específicas de las imágenes con que se va a trabajar, y más concretamente los objetos que se quieren detectar, es decir, las bayas de uva.

La estimación de las elipses es más fiable cuanto mejor sea la calidad de los bordes.

Detección de bordes

En la detección de bordes se tiene como punto de partida la imagen RGB que se convierte a escala de grises, utilizando una función propia 'rgb2gray' de MATLAB tal como se ha mencionado con anterioridad.



Figura 51. Imagen en escala de grises

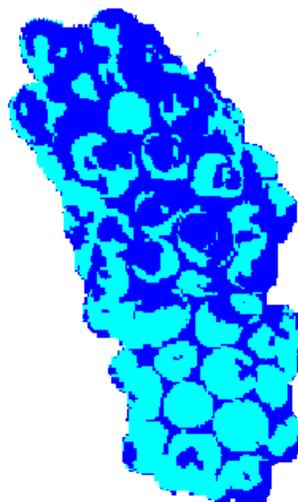
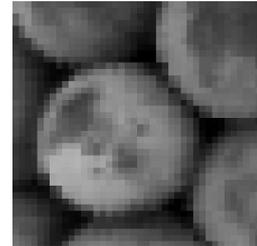
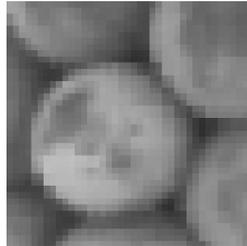
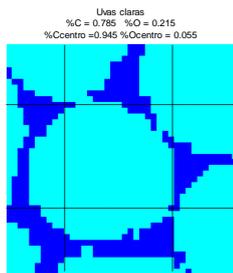


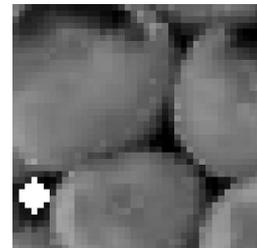
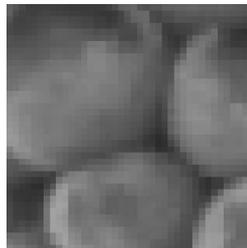
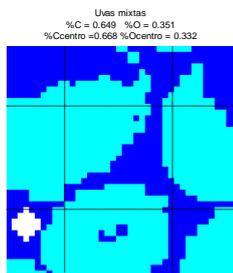
Figura 52. Tonalidad del racimo

En la imagen en escala de grises (Figura 51) se puede apreciar que existe una tonalidad de gris claramente distinta en las diferentes zonas del racimo, tal como se representa en la Figura 52, con lo que es posible clasificar las ventanas en: claras, oscuras y mixtas, según la tonalidad predominante. Ésta clasificación permite que el umbral de la detección se pueda adaptar mejor a las circunstancias de cada ventana, consiguiendo así mejorar el contraste de los bordes más relevantes, como se muestra en la Figura 53.

Ventana positiva



Ventana negativa



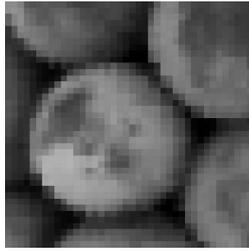
a.

b.

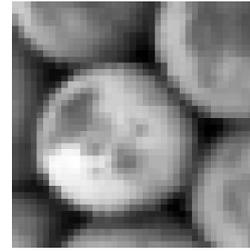
c.

Figura 53. Efecto de la umbralización por tonalidad de la ventana. a. Tonalidad. b. Imagen con umbral único. c. Imagen con umbral según tonalidad

Adicionalmente, como suele ser habitual, las imágenes presentan una cierta variabilidad (ruido), que sigue una distribución normal, en el valor de cada píxel que sumado al efecto de la reflectividad causada por la iluminación del entorno, hacen que a priori sea necesario el uso de un filtro paso bajo gaussiano. Sin embargo, se ha podido comprobar durante el desarrollo del trabajo que no es significativamente influyente (en la mayoría de casos), para la obtención del resultado final, por lo que se podría obviar.



a.



b.

Figura 54. Aplicación del filtro gaussiano sobre la imagen en escala de grises. a. Imagen original. b. Imagen con filtro gaussiano

A continuación, dentro de la gama de detectores de bordes de los que se dispone, el detector Canny es el que mejor resultados proporciona por sí solo (escogiendo un umbral de detección apropiado). Sin embargo, para tratar de mejorarlo sin tener que recurrir a desarrollar un nuevo detector, se ha optado por utilizar una combinación de detectores ya existentes. En este caso se optado por el detector Sobel y el gradiente.

El detector Sobel está definido por sus máscaras de detección horizontal (23) y vertical (24), a partir de las cuales se obtienen la variación horizontal (s_x) y vertical (s_y).

$$h_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \quad (23)$$

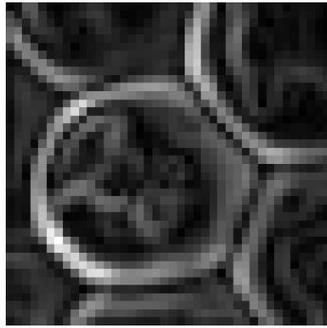
$$h_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \quad (24)$$

$$S = \sqrt{s_x^2 + s_y^2} \quad (25)$$

La magnitud del detector Sobel S (25) resultante se muestra en la Figura 55.a, sobre la cual se aplica el gradiente (26), tanto en sentido vertical (g_y) como horizontal (g_x), utilizando la función 'gradient' de MATLAB con $\Delta píxeles_x = \Delta píxeles_y = 1$, con lo que se obtiene el valor que interesa, la magnitud G , que viene dada por (27).

$$g_x = \frac{\Delta S_x}{\Delta píxeles_x}; g_y = \frac{\Delta S_y}{\Delta píxeles_y} \quad (26)$$

$$G = \sqrt{g_x^2 + g_y^2} \quad (27)$$



a.



b.

Figura 55 Combinación detector Sobel y su gradiente. a. Magnitud de la imagen con el Detector Sobel (S). b. Magnitud del gradiente aplicado sobre la imagen sobel (G), representado con una escala binaria.

El resultado de la combinación del detector sobel y su gradiente se muestra en la Figura 55.b, donde se ha utilizado, solo a modo ilustrativo, una escala binaria, que sirve para dilucidar más claramente los píxeles que van a ser considerados, a priori, como bordes.

Los píxeles de borde que se han detectado se pueden clasificar según la magnitud y dirección de su gradiente. La magnitud sirve para determinar si la variación, que ha llevado a considerar cada píxel como parte del borde, es significativa, pudiendo descartar aquellos cuyo valor esté por debajo de un cierto umbral. El umbral se define como la media de la magnitud de todos los píxeles con magnitud (G) mayor a cero. La dirección permite distinguir la orientación que tiene un borde, lo que permite cualificar la coherencia de la posición que tiene un borde en la imagen (ventana). Por ejemplo, no tiene ningún sentido que haya un borde horizontal (dirección de gradiente vertical) en los laterales de la imagen.

En primera instancia se distinguen en 3 tipos de píxeles¹²:

1. Píxeles con gradiente horizontal
2. Píxeles con gradiente vertical
3. Píxeles con gradiente vertical y horizontal

¹² Se considera que el gradiente es horizontal si su componente vertical es significativamente inferior ($G_x > 3 G_y$). De forma análoga se define el gradiente vertical. Así mismo, si la magnitud G es muy baja (menor al 0.1 una vez que se ha normalizado la magnitud), se considera cero.

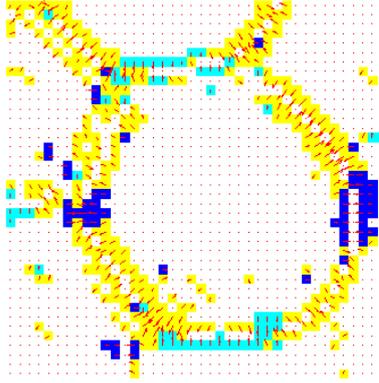


Figura 56 Clasificación de los píxeles. Azul horizontal, cian vertical y amarillo oblicuo. La zona blanca representa los píxeles de magnitud de gradiente cero. Las flechas rojas representa el gradiente (magnitud y dirección) en cada píxel.

Para la formación de los bordes se asigna a cada uno de los píxeles un tipo de borde según qué gradiente sea el preponderante dentro de una máscara (se escogió una de 3x3 centrada en el píxel en cuestión). Por ejemplo, en la Figura 57, el píxel se clasificaría como oblicuo, ya que entre los 9 píxeles de la máscara, la mayoría (4 píxeles) son de ese tipo.

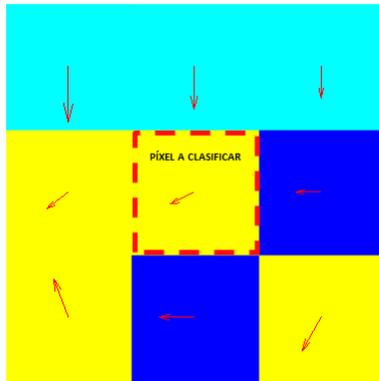


Figura 57 Matriz para clasificar píxeles de borde

Una vez clasificados cada uno de los píxeles, se agrupan según el tipo al que pertenecen, según la Tabla 3. La dirección de los píxeles se clasifica en 6 tipos, todos los píxeles contiguos del mismo tipo forman un borde. En la Figura 58 se muestran todos los bordes obtenidos para una ventana y se ha escogido un color para ilustrar cada tipo.

Tipo I	Tipo II	Tipo III	Tipo IV	Tipo V	Tipo VI

Tabla 3. Clasificación de los bordes según orientación

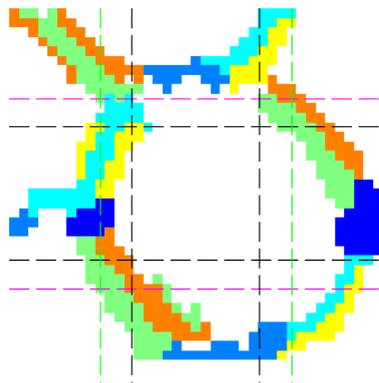


Figura 58 Bordes detectados en una ventana y coloreados por tipos

Una vez que se obtienen los bordes se aplican diversos criterios de selección, que se correspondería a las reglas que se obtendrían en una máquina de aprendizaje, para descartar aquellos que tengan menos probabilidad de ser bordes significativos o que sean inconsistentes para formar una elipse adecuada.

Para establecer la inconsistencia de los bordes, se realiza una clasificación adicional basada en la posición de los bordes (posición de su centro geométrico), dividiendo la ventana en zonas, como se muestra en la Figura 59, dependiendo del tipo de orientación del borde.

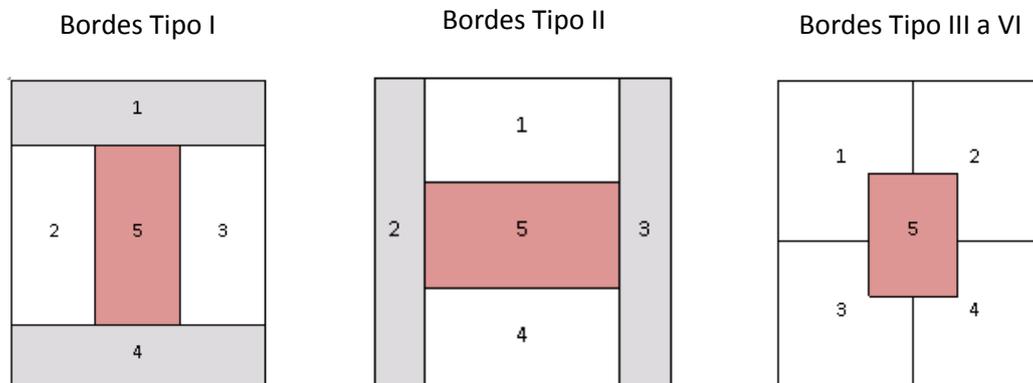


Figura 59. Clasificación de bordes por posición en la ventana

El conocimiento, a priori, de cómo deben ser y donde deben estar localizados los bordes que conforman una elipse dentro de la ventana, permite adoptar unos criterios de localización. De acuerdo a las posiciones definidas en la Figura 59, los bordes cuyo centro geométrico se ubique en zona gris, son descartados automáticamente, mientras los que lo hacen en la zona roja, dependiendo de si el borde está más o menos centrado o es más o menos fuerte¹³, el borde se descarta o no, esto se debe a que se está considerando que se van a detectar elipses más o menos centradas y que por tanto por el centro de la ventana no debería haber bordes significativos. Para el caso de los bordes Tipo III a VI, en la Figura 60 se resume las posiciones adecuadas que debería tener cada borde para ser válido. Entre corchetes se representa el tipo según el criterio de posición y sin corchetes el tipo según el criterio de orientación.

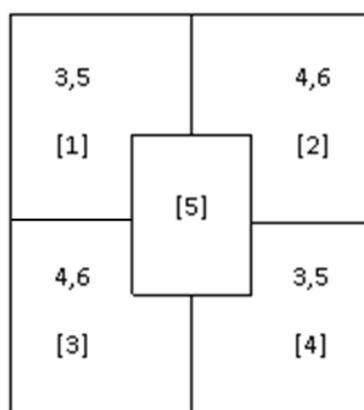


Figura 60. Mapa de posiciones para bordes tipo III, IV, V y VI

En ocasiones se pueden encontrar un número elevado de bordes válidos que definen la misma zona de una elipse. Para evitar trabajar con tantos bordes, ralentizando el cómputo y distorsionando la estimación de la elipse resultante, se aprovecha el hecho de que la uva tiene en sus bordes, en general, una tonalidad más oscura que en el resto de la uva, tal y como se observa en la Figura 53.c. Por lo tanto somos capaces de seleccionar aquellos bordes que tienen una intensidad media (de la escala de grises) más baja, es decir, los más oscuros.

Como criterio adicional, se puede considerar que los bordes de tipo III a VI (gradiente oblicuo) no deberían formar líneas horizontales ni verticales, es decir deberían presentar una cierta pendiente. Además, se han eliminado todos aquellos bordes que no tengan un mínimo número de píxeles.

Aplicando estos criterios de filtrado de bordes se obtienen finalmente unos bordes consistentes (Figura 61), los cuales son utilizados para la estimación de elipses.

¹³ Borde compuesto por un elevado número de píxeles.

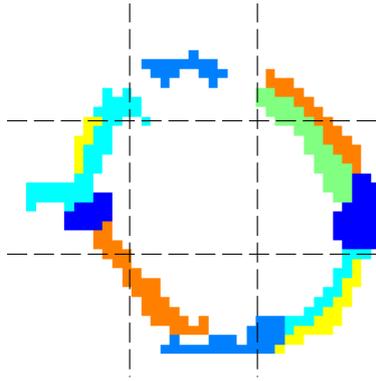


Figura 61 Bordes consistentes detectados en una ventana

Adicionalmente, se ha decidido descartar aquellas ventanas cuyo número de bordes (válidos) de pequeña longitud (menos o igual a 5 píxeles) represente más del 60% del total de bordes, ya que es un indicativo de que la variación de color es muy aleatoria y por tanto difícilmente podría haber una elipse alojada en la ventana en cuestión, o si la hay, lo más probable es que se trate de una baya poco definida y que por lo tanto la elipse estimada no sea apropiada.

Estimación de la elipse

La estimación de la elipse se realiza a través de una aproximación lineal, siguiendo el algoritmo Buckstein, que emplea:

4. La definición de la cónica

$$x'Ax + B'x + C = 0 \quad (28)$$

5. El error de aproximación, que se minimiza por mínimos cuadrados, con la restricción lineal (29), siendo λ_i los autovalores de A

$$\lambda_1^2 + \lambda_2^2 = 0 \quad (29)$$

Una aproximación no lineal proporciona una elipse, por lo general, más precisa. Sin embargo, el tiempo de cómputo es significativamente mayor, y al ser un método iterativo puede no converger. Además presenta serios inconvenientes cuando la elipse detectada es próxima a ser una circunferencia, en cuyos casos se deja de utilizar la aproximación no lineal y se pasa a utilizar la aproximación lineal, lo cual es bastante frecuente dada la morfología de las bayas de uva, de ahí que se justifique la utilización de solo la aproximación lineal. En la Figura 62 se muestra un ejemplo de estimación de una elipse a partir de los bordes que han pasado el filtrado previo.

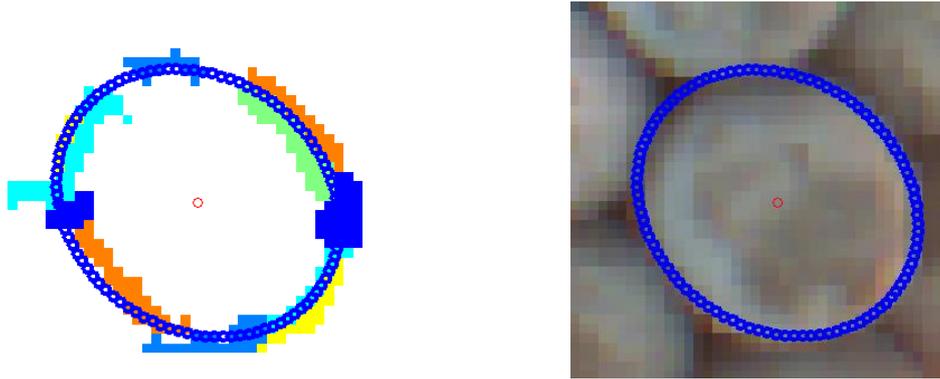


Figura 62. Detección de la elipse en el Algoritmo Prasad

Recorriendo con la ventana toda la imagen se detectan las elipses de la Figura 63.a. A priori, las elipses de uvas (detectables) se deben detectar en más de una ventana, es decir, se considera que una elipse es incorrecta si solo se ha detectado una vez y por tanto se descarta. Siguiendo éste criterio, se agrupan el resto de elipses de acuerdo a la distancia euclídea entre sus centros, formándose los grupos de la Figura 63.b. Finalmente, en cada uno de los grupos de elipses se estima una elipse resultante, las cuales se observan en la Figura 63.c.

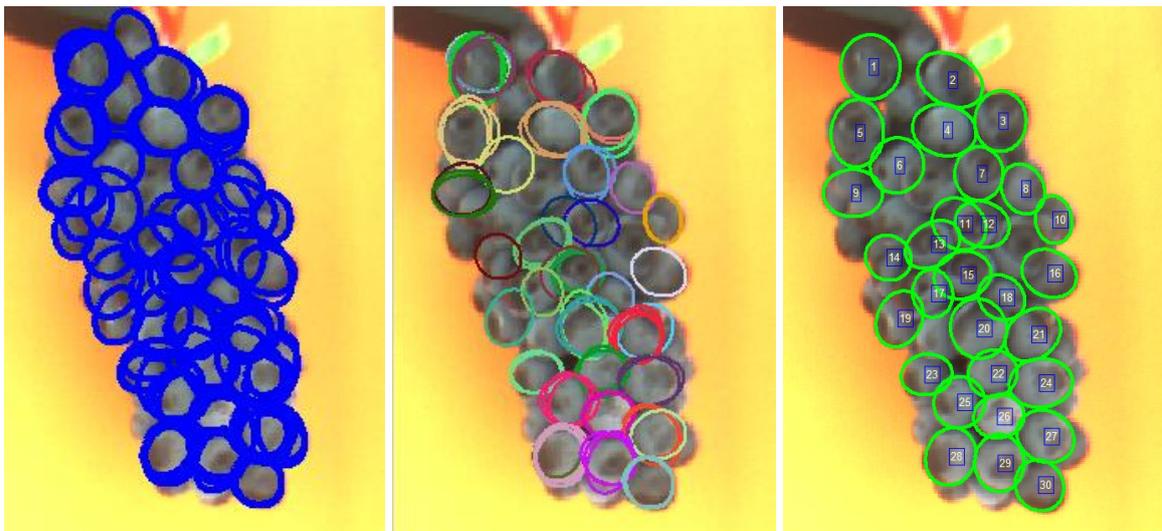


Figura 63. a. Elipses detectadas en todo el racimo con la primera propuesta. b. Agrupamiento de las elipses. c. Elipses resultantes de cada grupo

3.3.2. Ventajas e inconvenientes

El número de elipses detectadas elevado, el relativamente bajo coste computacional y la utilización de técnicas bastante conocidas en su implementación son las ventajas reseñables.

Sin embargo, presenta algunos inconvenientes a resaltar como: el elevado número de parámetros a definir, tanto para la detección de las elipses como para el descarte de elipses

mal definidas, lo que lo hace poco robusto. Aunque, considerando que se ha ido aprendiendo cómo establecer parámetros, no debería resultar difícil modificarlos.

Este algoritmo se ha propuesto atendiendo las necesidades específicas del trabajo, por lo que se da por sentado que los racimos, en general, cumplen con ciertas características de morfología, por lo que si se sale demasiado de dichas premisas los resultados va decayendo en validez.

3.4. Conclusiones de la detección de elipses

Se han presentado tres propuestas de solución para la detección de elipses, uno basado en la información de la curvatura de los bordes y otro en la obtención de un modelo que define el objeto a detectar, además de una propuesta propia que puede constituir una solución híbrida entre las otras dos propuestas ya que, adopta algunas ideas tanto del uno como del otro.

Decidir cuál de las propuestas es la mejor, no es una tarea sencilla; sin embargo, sí se puede seleccionar la opción más adecuada según las circunstancias que se presenten.

En ese sentido se puede decir que cuando se tengan imágenes en las cuales los objetos elípticos a detectar estén bien definidos la utilización del algoritmo de curvatura es sin duda la solución más rápida de implementar, ya que es posible detectar elipses de distintas variedades.

Por otro lado, si se tiene el tiempo necesario que se requiere para entrenar un modelo de detección, sería mejor que la variabilidad de los objetos a detectar no fuera muy elevada, ya que el algoritmo basado en modelos brinda la posibilidad de hacer una buena detección siempre que el modelo obtenido se aproxime, en forma, al objeto a detectar.

Por último, si el objeto está bien definido y la variabilidad es más bien poca se puede recurrir a una solución utilizando ideas de las otras dos propuestas, donde la gran baza, a priori, está en la utilización del conocimiento sobre el comportamiento de los objetos y la forma en que pueden ser detectados. Siempre buscando de alguna forma se supere en algún aspecto (número de elipses detectadas, tiempo de cómputo, ratio de falsos positivos,...) que se considere de especial interés. Es decir, ésta constituiría una solución ad hoc, con todas las ventajas e inconvenientes que eso conlleva.

Por lo tanto, dado que se va a tratar con imágenes en donde los objetos de interés no están excesivamente bien definidos, y que el número de variedades de racimos de uva es medianamente alto (10-11 variedades) pero sobretodo difieren en aspectos como el color o tamaño, hace que la mejor forma de solucionar el problema de detección sea la que se basa en modelos.

Cabe mencionar, que si se quisiera resolver el problema para una variedad en concreto, o más bien para una tipología de racimos, la mejor opción para conseguirlo sería con la propuesta propia que se ha presentado.

3.5. Búsqueda de correspondencias.

La búsqueda de correspondencias que se plantea, utiliza algunos conceptos empleados en el caso de los círculos, en cuyo caso se hará una breve mención, mientras que aquellos conceptos que difieran en algún aspecto relevante o no se han utilizado en la detección de círculos se desarrolla con mayor detalle.

En el caso de los círculos se había planteado el problema para la imagen al completo, en cambio ahora se selecciona una región de interés (ROI) que englobe todo el racimo, además de cumplir unos criterios particulares que se han definido con el fin de facilitar la búsqueda de correspondencias.

La ROI es un área rectangular definida por sus vértices: x_1, x_2, y_1 e y_2 . La ROI de la imagen izquierda y derecha deben cumplir la restricción epipolar, por tanto $y_{1izq} = y_{1der}$ y $y_{2izq} = y_{2der}$. Las dimensiones de la ROI se definen a partir del racimo segmentado, como los de la Figura 8, añadiendo un margen de holgura de unos 20 píxeles para asegurar que las uvas de los extremos se ubiquen, en su momento, centradas en su correspondiente ventana. Para cumplir la restricción epipolar se toma $y_1 = \min(y_{1izq}, y_{1der})$ e $y_2 = \max(y_{2izq}, y_{2der})$.

Adicionalmente, se incluye el criterio de simetría, con el que se pretende que el eje de simetría vertical del racimo esté centrado relativamente a la ROI, ya que es uno de los supuestos utilizado en el emparejamiento de elipses. El ancho de la ROI (valores de x_1 y x_2) se determina de forma similar que y_1 e y_2 , introduciendo el valor $d = x'_{izq} - x'_{der}$ para salvaguardar la diferencia relativa de posición del racimo en el par de imágenes. Por lo tanto, se tiene que $x_1 = \min(x_{1izq} - d, x_{1der})$ y $x_2 = \max(x_{2izq} - d, x_{2der})$.

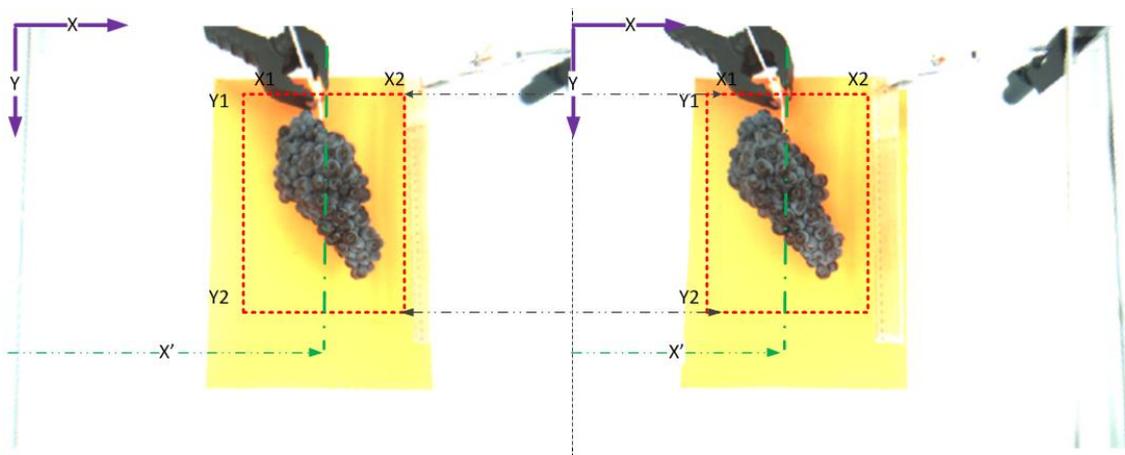


Figura 64. Selección de la región de interés (ROI) en el par estéreo de imágenes

La selección de máximo y mínimo se ha hecho considerando que los valores x_1, x_2, y_1 e y_2 están definidos según el sistema de coordenadas de imagen, cuyo origen está en la esquina superior izquierda de cada imagen, tal como se observa en la Figura 64.

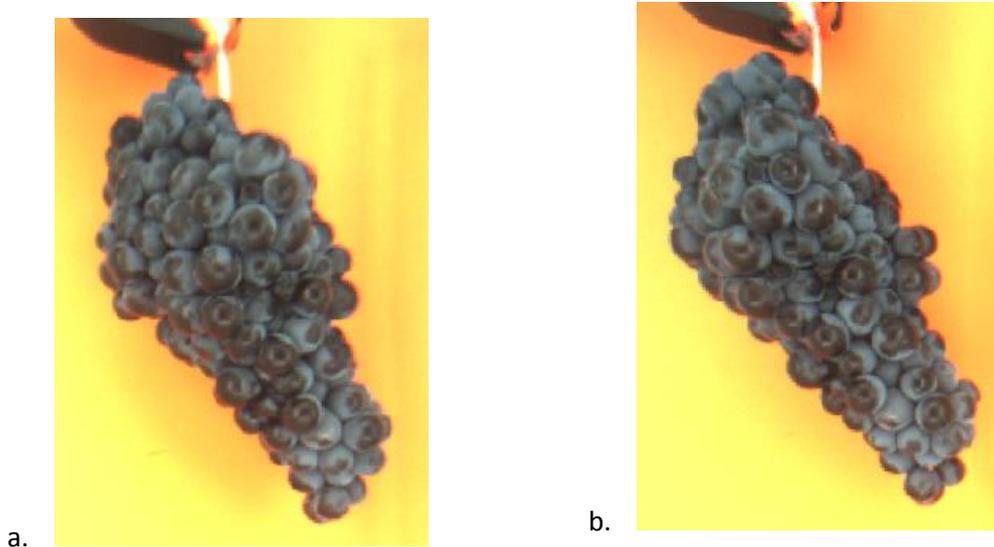


Figura 65. ROI de las imágenes izquierda (a) y derecha (b) de los racimos

El primer criterio de emparejamiento es el de la geometría de las elipses. Por un lado está el centro de las elipses, que sigue de forma idéntica la secuencia de implementación aplicada en el caso de los círculos, se escoge una elipse (elipses base) de la imagen izquierda y se obtiene sus elipses candidatas en la imagen derecha. Por otro lado, el tamaño, que no se tenía en cuenta para círculos porque se comparaban círculos del mismo radio. Se compara los puntos máximos y mínimos de la elipse base y sus elipses candidatas, si la suma de las diferencias (en valor absoluto) de alguna de las elipses candidatas es superior a 8 píxeles, entonces se descarta. Éste umbral es subjetivo, y se ha ido adaptando según se ha ido probando sobre imágenes.

Considerando la simetría de las imágenes, se compara la posición horizontal del centro de la elipse base y sus candidatas, de forma que se escoge aquella pareja que menor valor de discrepancia tenga, es similar al criterio de la disparidad utilizado en los círculos. Para más detalles, ver Anexo 6.1.

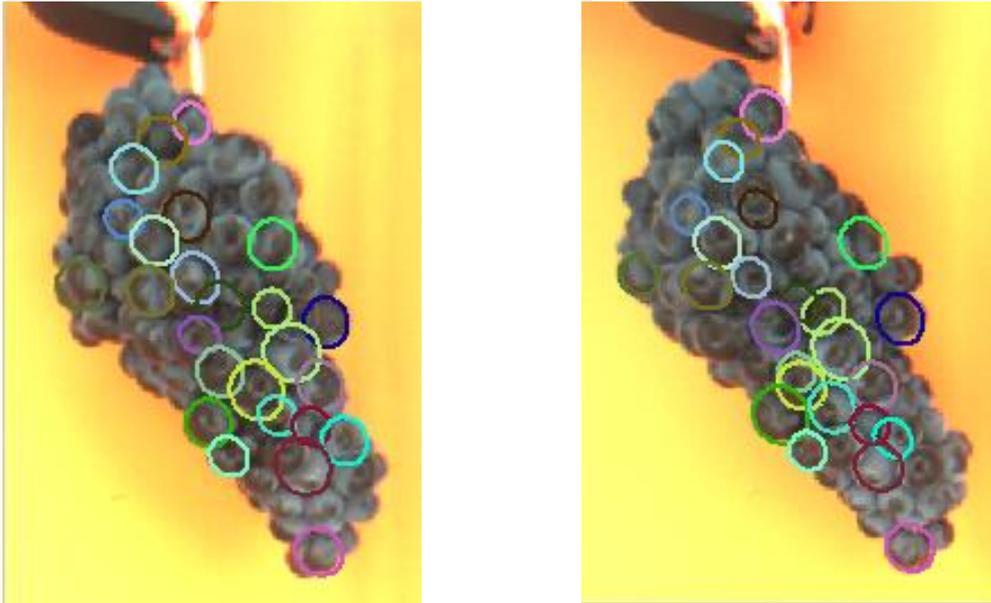


Figura 66. Emparejamiento de elipses

3.6. Reconstrucción y estimación de tamaño

La reconstrucción y estimación del tamaño de baya se hace de forma idéntica a la descrita para el caso de los círculos, con la única excepción, de que al tratarse de elipses habrá cuatro posibles soluciones, de las cuales se escogerá una sola.

La razón para que hayan cuatro soluciones es que el algoritmo de reconstrucción empleado tiene como hipótesis que las proyecciones en las imágenes son círculos; por lo tanto, se aproxima la elipse a un círculo de radio igual al semieje mayor o al semieje menor, de ahí que existan dos opciones tanto para la imagen izquierda como para la derecha.

La elección de la mejor solución se hace en base a la diferencia de área que hay entre la proyección elíptica y los círculos generados. El círculo cuya diferencia de área sea menor es la que se escoge como proyección para la reconstrucción de la baya.

En la Figura 67 se muestran ejemplos de los círculos generados para las proyecciones elípticas (elipses detectadas y emparejadas), tanto para la imagen izquierda como derecha. En la Figura 68 se representan las proyecciones de imagen de la esfera que se ha generado y las proyecciones detectadas.

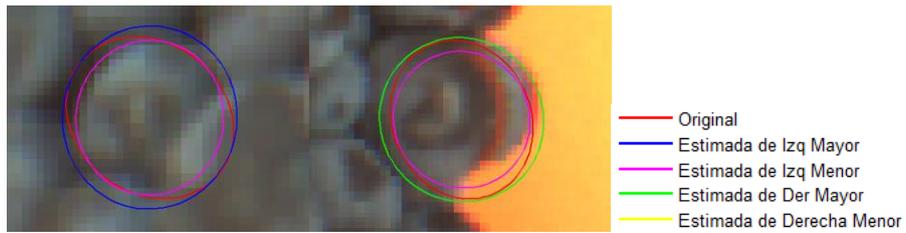


Figura 67. Aproximación de una elipse a un círculo

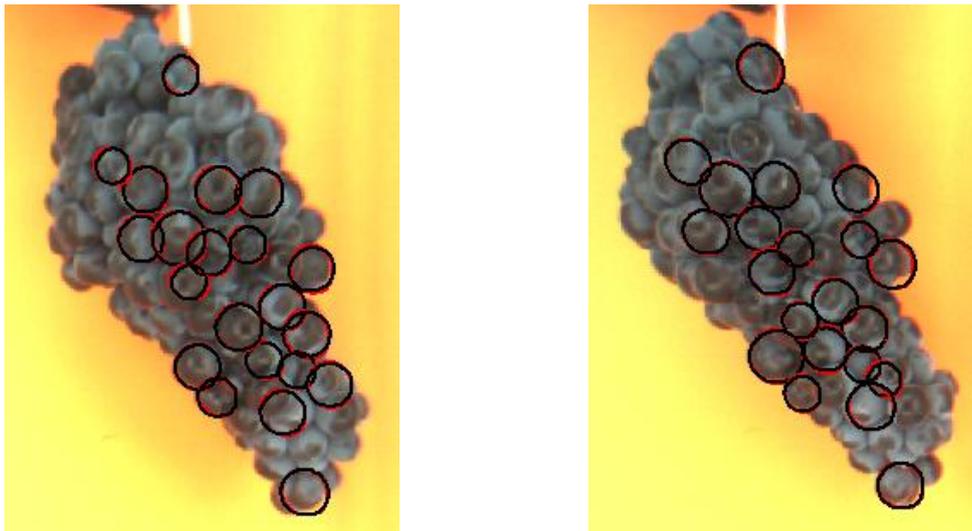


Figura 68. Selección de la mejor alternativa de solución. Rojo proyección original. Negro mejor proyección

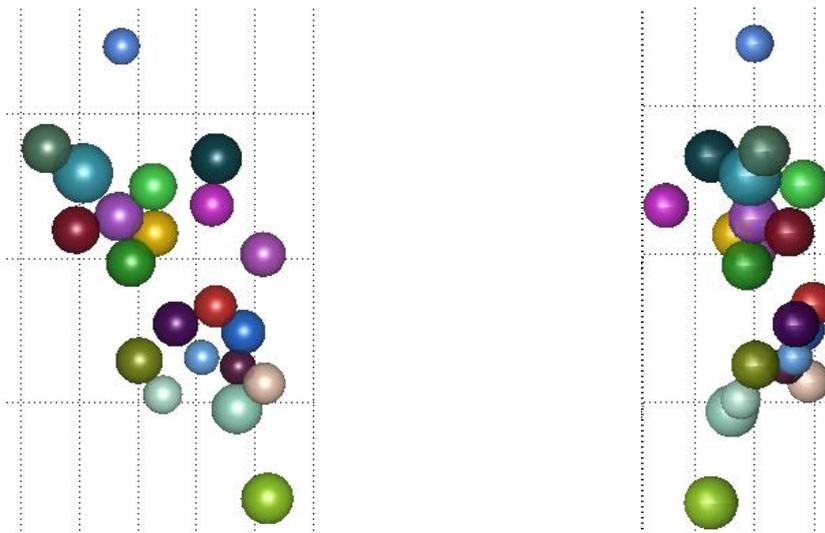


Figura 69. Representación 3D del racimo de uvas. (Izquierda) Alzado. (Derecha) Perfil

3.7. Resultados

Todo el proceso descrito anteriormente se prueba sobre un conjunto de 10 racimos para cada una de las clases de uva de las que se dispone. En la Tabla 4 se muestran el número de elipses detectadas así como el radio estimado para cada uno de los racimos. En la Tabla 5 se muestra los errores de estimación tanto en la posición espacial de las bayas (posición del centro), así como el error del tamaño (radio) de la baya.

		Bobal		Cabernet Franc		Cinsaut	
N	N. elipses	R (mm)	N. elipses	R (mm)	N. elipses	R (mm)	
1	23	8,376489	12	7,062297	27	8,107978	
2	23	8,145304	17	7,900821	28	8,420643	
3	23	7,969164	21	7,529889	22	8,376232	
4	21	7,950205	13	7,498821	25	8,179042	
5	27	8,248551	18	7,51911	20	7,868937	
6	23	8,15167	6	7,866046	26	8,378971	
7	35	7,654868	13	6,725628	17	8,665593	
8	35	7,599965	7	7,347557	15	7,986061	
9	37	8,135292	11	6,754807	15	8,544619	
10	37	7,376082	20	7,024741	19	8,693377	
		Miguel de Arco		Monastrell		Moravia Agria	
N	N. elipses	R (mm)	N. elipses	R (mm)	N. elipses	R (mm)	
1	33	7,412284	15	7,018477	13	7,691464	
2	19	7,456943	19	7,671172	18	7,708614	
3	16	7,206527	17	8,356627	15	7,37667	
4	22	7,725354	23	7,946674	13	8,041846	
5	15	7,917614	12	7,019114	12	7,654906	
6	21	7,564738	17	7,491163	10	6,710249	
7	18	7,944578	15	7,545136	8	8,38652	
8	15	8,426976	24	7,818951	16	7,857002	
9	20	7,416991	17	7,661687	20	7,370502	
10	15	7,227039	12	8,281228	21	8,081338	
		Moscatel Morisco		Mouraton		Naparo	
N	N. elipses	R (mm)	N. elipses	R (mm)	N. elipses	R (mm)	
1	31	7,596001	16	8,272013	30	8,113331	
2	26	7,516079	20	8,55875	38	8,312899	
3	27	7,563072	21	8,473577	36	8,023607	
4	35	7,711675	16	8,295595	35	7,896064	
5	32	7,395904	18	7,870801	34	7,78884	
6	23	8,013268	15	7,15291	18	8,386437	
7	34	7,489814	22	7,762228	21	7,890621	
8	31	8,334559	17	7,633125	30	8,127891	
9	38	7,427982	19	7,522963	41	8,233478	
10	31	8,073663	19	7,885042	43	7,582673	
		Planta Mula		Ruby_Seedless			
N	N. elipses	R (mm)	N. elipses	R (mm)			
1	29	7,947314	42	7,629351			

2	28	7,792171	26	8,404564
3	20	7,853888	39	8,00431
4	23	7,679594	38	8,181142
5	25	7,501346	34	7,675032
6	35	7,417818	22	7,788469
7	26	7,311377	35	7,923916
8	30	7,812275	34	8,421026
9	37	7,942341	37	7,492705
10	44	7,746517	30	7,627047

Tabla 4. Resultados experimentales. Detección de elipses en 10 racimos para cada una de las variedades de uva disponibles. Número de elipses detectadas y radio estimado.

	Bobal	Cabernet Franc	Cinsaut	Miguel Arco	Monastrell
error posición (mm)	3,309652	3,734618	2,851800	4,207858	4,247777
error radio (mm)	0,61512	1,857818	0,744338	1,072955	0,808853
	Moravia	Mouraton	Naparo	Planta Mula	Ruby
error posición (mm)	3,978078	4,130936	2,953322	2,740468	2,720855
error radio (mm)	1,440295	1,563717	0,625412	0,577511	0,846265

Tabla 5. Resultados experimentales. Comparación de los modelos 3D obtenidos con los modelos 3D patrón de los racimos disponibles en la detección de elipses

4. Conclusiones finales

Se han presentado varias alternativas para la reconstrucción 3D de racimos de uva, que son utilizadas para la estimación del tamaño de baya de uva. Por un lado está la que se basa en la detección de círculos y por otro lado la que lo hace en base a detección de elipses.

El número de proyecciones obtenidas con detección de círculos es inferior a la que se obtiene con detección de elipses. La precisión en la detección en posición y radio es mejor en círculos que en elipses, aunque en cuanto a la estimación del tamaño de baya son similares.

Una de las razones para que el error de detección de elipses sea más alto que en círculos es que el algoritmo de reconstrucción utilizado tiene como hipótesis que las proyecciones sean circulares, por lo que para el caso de elipses se toma como valor de radio uno de los semiejes, siendo este error el que se va arrastrando.

Como conclusión final, se puede decir que la reconstrucción basada en detección de círculos es superior a la que se basa en elipses, siempre que, los objetos a detectar sean lo más esféricos posibles, como sucede en la mayoría de bayas de los racimos con los que se ha tratado. Por lo tanto, cuando menos esféricas sean las bayas o más bien sus proyecciones en imagen, peor será la reconstrucción basada en círculos y mejor será la que se basa en detección de elipses. En el Anexo 6.3 se muestran unas imágenes de comparación que corroboran esta conclusión.

5. Trabajos futuros

En el sentido de dar solución o mejorar lo conseguido hasta ahora, como tareas a futuro se plantea la detección de un mayor número de elipses, de las no detectables, es decir, aquellas que, en este trabajo, han sido descartadas por falta de métodos para tratarlas.

Las elipses no detectables son aquellas de las que se posee poca información debido a los ocultamientos que se producen, en ese sentido se tienen principalmente dos casos: aquellas bayas que aparecen, en poca cantidad, pero en ambas imágenes estereó, y, aquellas donde el ocultamiento es tan excesivo que la baya aparece proyectada solo en una de las imágenes.

Por lo tanto se deja para futuros proyectos el planteamiento interactivo de la reconstrucción del racimo de uvas, partiendo de la solución preliminar obtenida en este trabajo y generando hipótesis de baya que permitan completar en la mayor cantidad y con la mayor precisión posible todas las bayas del racimo.

6. Anexos

6.1. Búsqueda de correspondencias para Elipses

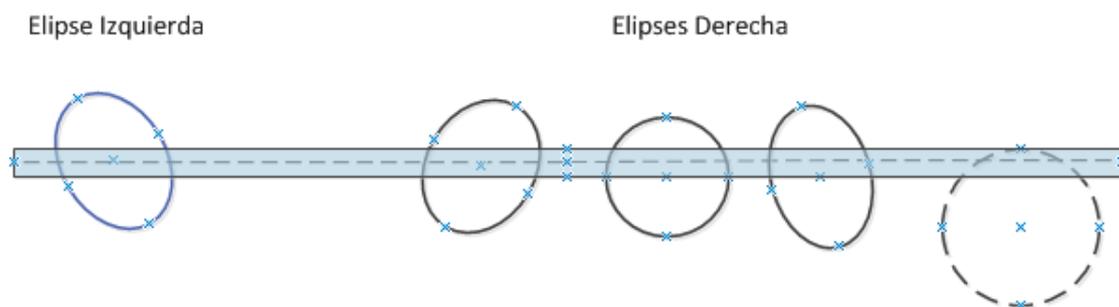
El objetivo es emparejar elipses de la imagen izquierda con su correspondiente en la imagen derecha, de forma que ambas constituyan las proyecciones de una esfera, que es la que representa la baya de uva.

La búsqueda de correspondencia se puede hacer tanto partiendo de una elipse en la imagen izquierda y buscando su homólogo en la imagen derecha, o viceversa. Lo óptimo sería hacerlo en los dos sentidos y quedarse con aquellas parejas que estén mejor emparejadas, definiendo algún tipo de error de emparejamiento. Sin embargo, se ha podido observar durante la implementación del algoritmo que no hay una diferencia significativa en el resultado obtenido; por lo tanto, se ha decidido solo hacer una búsqueda de izquierda a derecha y utilizar esas parejas de elipses en la reconstrucción. A continuación se describe todos los pasos que se han establecido para definir las parejas.

- Selección de candidatos en una banda de altura definida

Los primeros candidatos se obtienen definiendo una banda de una semi-altura 5 píxeles, valor que se ha escogido luego de hacer unos cuantos ensayos, distribuidos simétricamente arriba y abajo. El ancho de la banda cubre todo el ancho del racimo, o lo que es lo mismo, no hay límite de ancho. Todas las elipses de la imagen derecha cuyo centro cae dentro de la banda, se convierten en elipses candidatas, las que no cumplen este requisito son descartadas.

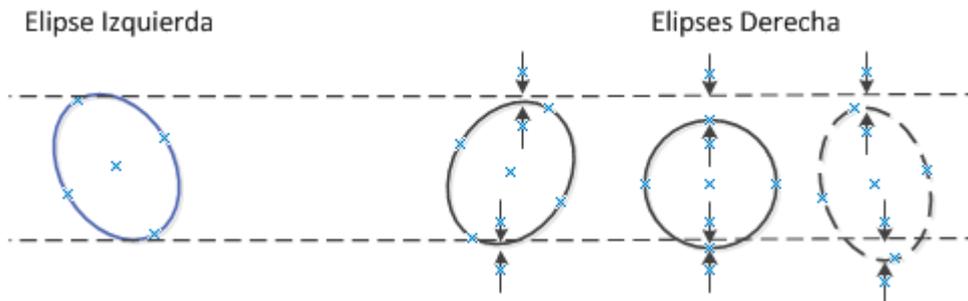
$$\left| C_{Y-IZQ_j} - C_{Y-DER_i} \right| < 5 px \quad (30)$$



- Selección de candidatos según las dimensiones geométricas.

Una vez que se han descartado las elipses que no pertenecen a la banda, se compara los puntos más altos y más bajos de la elipse base con las elipses candidatas. Como criterio se ha escogido que la suma de la diferencia (en valor absoluto) de las crestas y de los valles, no supere los 8 píxeles, de ser así se descarta esa elipse.

$$\left| Cresta_{Y-IZQ_j} - Cresta_{Y-DEr_i} \right| - \left| Valle_{Y-IZQ_j} - Valle_{Y-DEr_i} \right| < 8 px \quad (31)$$

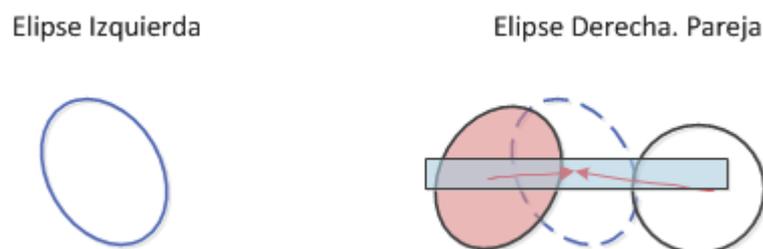


- Búsqueda en banda de anchura definida

Teniendo en cuenta que se han escogido las ROI, de forma que el racimo se localice de forma simétrica (según su eje vertical), esto quiere decir que el centro de la elipse derecha que se empareja con la elipse izquierda en cuestión, debería estar localizada en las inmediaciones de la misma posición horizontal del centro de la elipse izquierda.

Debido a la disparidad, el ancho de la banda que se define para buscar la candidata final no es tan estrecho como ocurría en el caso de la altura de la banda utilizada en la primera búsqueda. Idealmente el ancho debería ser variable, ya que la disparidad varía en función de la profundidad respecto de la cámara que se encuentre el racimo, pero como ese dato no lo disponemos sino hasta cuando se está haciendo la reconstrucción 3D, por lo que no tendría demasiado sentido, utilizar una información de la que no se dispone. Por lo tanto se ha optado por definir un semi-ancho de banda de 20 píxeles, centrado en la misma posición del centro de la elipse izquierda.

La candidata definitiva es aquella cuya distancia euclídea entre centros es la menor. Siendo por tanto ésta pareja de elipses, las proyecciones utilizadas en la reconstrucción 3D de su baya correspondiente.



6.2. Mejoras en la detección

El algoritmo pedestrians presentado incluye, originalmente, un escalado de la máscara. Éste escalado es utilizado con el fin de detectar personas de diversos tamaños, pero que sigan teniendo las características que las definen como personas. En el caso de los racimo de uvas, las características son la magnitud y orientación del gradiente, para ellas se ha creado una máscara, que es la que se ha estado utilizando, y es la que se va a escalar.

Las bayas tienen un tamaño similar, es decir, varía dentro de un rango pequeño de valores de semieje. Este pequeño rango de valores se traduce en la utilización de 3 tamaños de máscara: pequeña, mediana (por defecto) y grande. Lo ideal hubiese sido realizar el entrenamiento con uvas para cada uno de estos tamaños; sin embargo, se ha optado, por sencillez, utilizando un factor de escala sobre la máscara inicial. En la Figura 70 se muestra un ejemplo del escalado de la máscara, en concreto para la máscara de probabilidades de magnitud, para el resto de máscaras se haría lo mismo, obteniendo finalmente el modelo escalado con el factor de escala correspondiente.

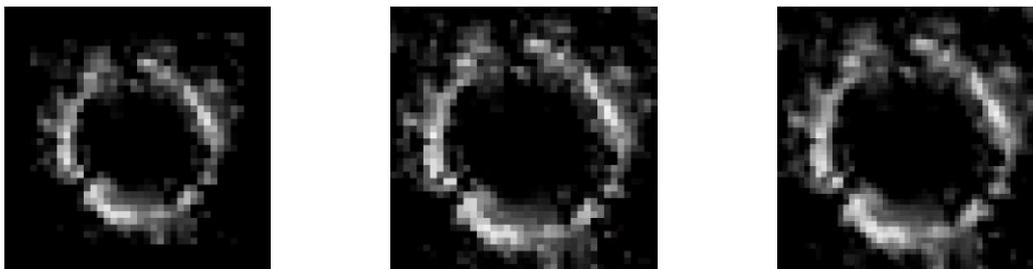


Figura 70. Máscara de probabilidades escalada de magnitud. a. Pequeña 0.8. b. Mediana 1. c. Grande 1.2.

De los 3 tamaño de uva definidos, el mediano es el más frecuente (por eso es el valor por defecto), mientras que los tamaños pequeño y grande son menos probables. Esto ocurre en general, aunque hay clases de uvas que presentan un tamaño de baya predominantemente pequeño o grande, en cuyo caso se utilizaría la máscara correspondiente como la principal.

Debido a que las variedades de uva están ya definidos (10 clases), para cada una de ellas se ha establecido su tamaño predominante, el cuál en su mayoría es el tamaño estándar y tan solo hay una variedad cuyo tamaño es pequeño y otra cuyo tamaño es grande, que puede incluso ser considerado como de tamaño estándar. Por lo tanto, a continuación se muestra la secuencia para racimos de tamaño estándar, la cual será análoga para los otros tamaños, cambiando en orden de aplicación de los modelos.

Las imágenes de racimo de uva que se utilizan son bastante complejas de tratar, debido a la gran cantidad de bordes indeseables que se generan, es decir, todos aquellos bordes que no forman el contorno de alguna baya. Por lo tanto el número de elipses detectada utilizando el modelo de baya pequeña suele ser elevado y/o tener un ratio de falsos positivos bastante significativo. En el sentido opuesto se encuentran las bayas grandes, las cuales se suelen

detectar en pequeño número, pero su ratio de falsos positivos no es elevado. En la Figura 71 se muestran las elipses detectadas, que no emparejadas, para los distintos tamaños, donde se comprueba lo mencionado anteriormente.

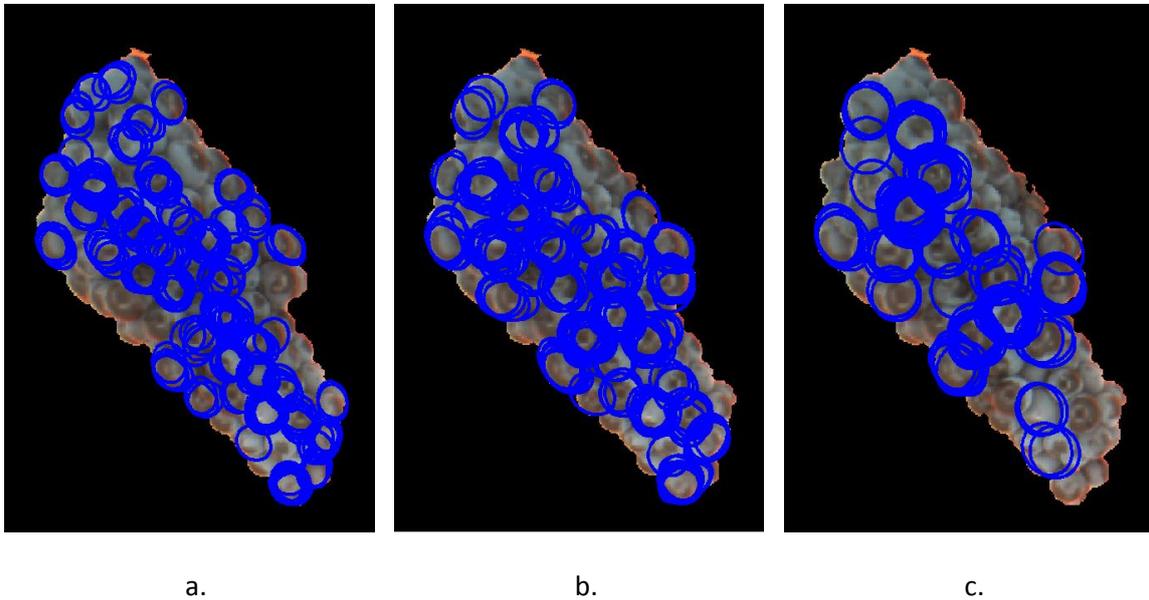


Figura 71. Detección de elipses con las máscaras escaladas. a. Pequeña. b. Mediana. c. Grande

Teniendo en cuentas éstos resultados, se ha optado por hacer una combinación de ellos, para conjuntar tanto el número de elipses detectado como el ratio de falsos positivos (Figura 72.a). Todo este conjunto de elipses sirve para dilucidar las proyecciones que se utilizan para la reconstrucción 3D del racimo, y tal como se ve en la Figura 72.b hay todavía un área significativo de racimo donde no se han detectados elipses.

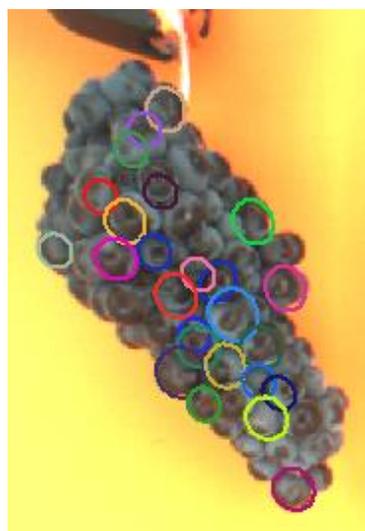
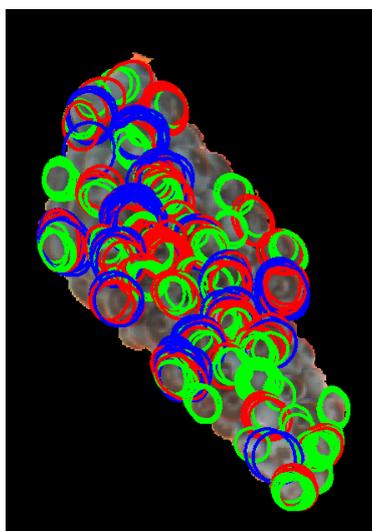


Figura 72. Combinación de elipses detectadas para los tres tamaños. a. Elipses totales detectadas. b. Elipses que son utilizadas en la reconstrucción 3D.

Por lo tanto, se propone incluir una detección secuencial, utilizando la zona del racimo donde no se ha detectado elipses que sean proyecciones, es decir, del racimo se van restando el área que ocupan las proyecciones utilizadas y sobre ésta se vuelve a aplicar la detección de elipses (Figura 73). Este proceso se repite hasta que no haya más área de racimo que detectar o se llegue a un límite de iteraciones, que se ha fijado en 10, para salvaguardar que en alguna ocasión se produzcan bucles infinitos.

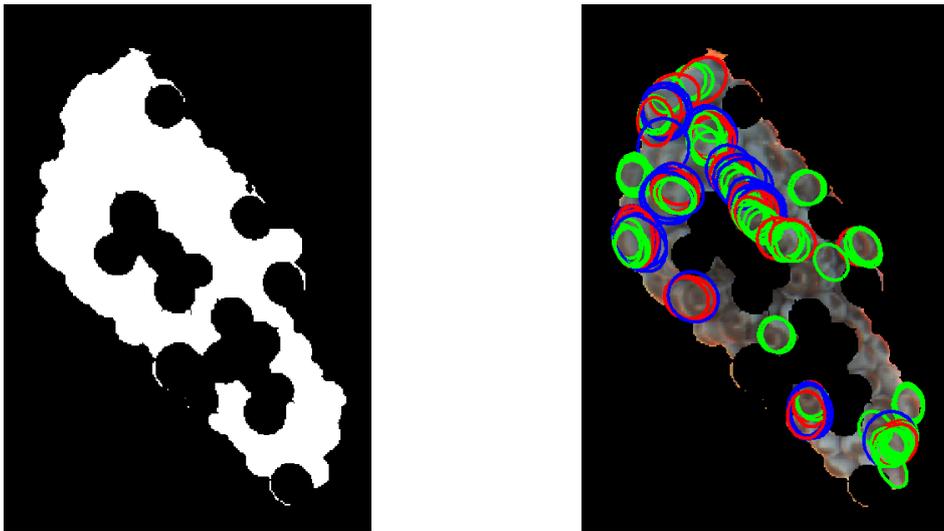


Figura 73. Secuencialización de la detección.

Una de las limitaciones que se ha detectado es que, en caso de que las elipses que se vayan detectando no sean demasiado precisas, la detección de las mismas en las siguientes iteraciones puede verse afectado hasta tal punto de perder alguna elipse con bordes aparentemente bien definidos.

6.3. Círculos vs. Elipses

Aquí se muestran algunos ejemplos de imágenes de trozos de racimos que sirven para dilucidar el alcance que tiene la reconstrucción 3D por círculos y por elipses.

Las bayas de uva son objetos cuasi-esféricos, por lo que su proyección en imagen pueden ser círculos o elipses. En la Figura 74 se muestra como las uvas cuya proyección son más circulares son detectadas con más precisión con el método de detección de círculos, siempre que no influya demasiado efecto de la componente especular. La detección de elipses por otra parte es más precisa y/o es capaz de detectar aquellas proyecciones que menos circulares sean.

Se puede decir por lo tanto que, la detección por elipses es más potente en cuanto a detección pero menos precisa, además de ser más tolerable a al efecto de la componente especular así como a los ocultamientos.

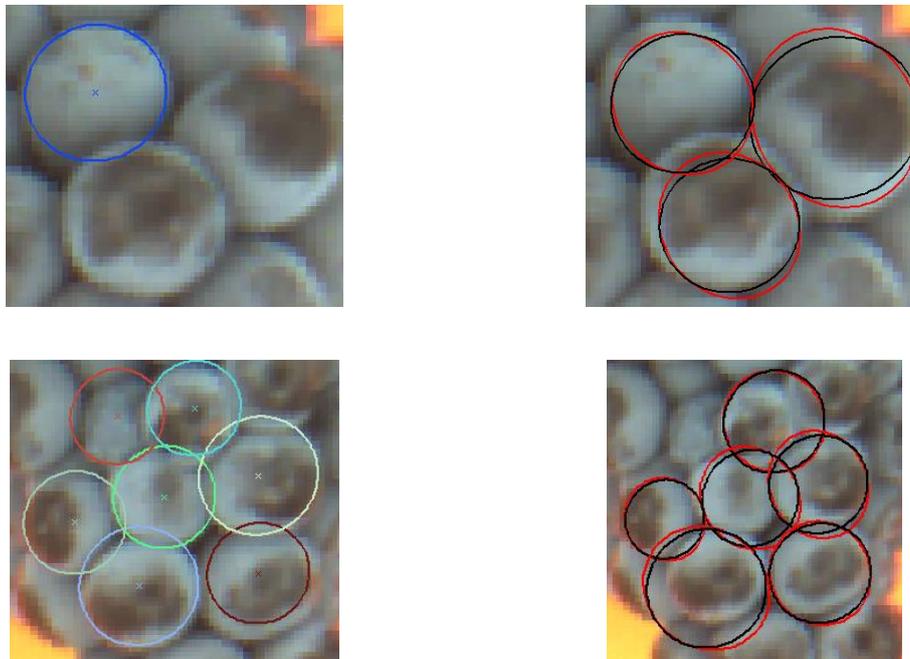


Figura 74. Comparación de detección de uvas con detección por círculos (izquierda) y por elipses (derecha)

7. Referencias

- 1] R. Rocher, K. Herzog, A. Kunkel, A. Kicherer, R. Töpfer and W. Förstner, "Automated image analysis framework for high-throughput determination of grapevine berry sizes using conditional random fields.," *Computers and Electronics in Agriculture*, no. 100, pp. 148-158, 2009.
- 2] S. Verdú, E. Ivorra, A. J. Sánchez, J. Girón, J. M. Barat y R. Grau, «Comparison of TOF and SL technics for in-line measurement of food item volume using animal and vegetable tissues.,» *Food Control*, nº 33(1), pp. 221-226, 2013.
- 3] P. Udomkun, M. Nagle, B. Mahayothee y J. Müller, «Laser-based imaging system for non-invasive monitoring of quality changes of papaya during drying,» *Food Control*, nº 42(0), pp. 225-233, 2014.
- 4] B. Cyganek y J. P. Siebert, «An introduction to 3D computer vision techniques and algorithms,» *John Wiley & Sons*.
- 5] D. Young, «Matlab Central. MathWorks Inc,» 2010. [En línea]. Available: <http://www.mathworks.com/matlabcentral/fileexchange/26978-hough-transform-for-circles>. [Último acceso: 16 07 2014].
- 6] C. Y. Wong, S. C. F. Lin, T. R. Ren y N. M. Kwok, «A Survey on Ellipse Detection Methods,» *IEEE International Symposium on Industrial Electronics (ISIE)*, pp. 1105-1110, 2012.
- 7] D. K. Prasad y M. K. H. Leung, «Methods for Ellipse Detection from Edge Maps of Real Images,» de *Machine Vision - Applications and Systems*, D. F. Solari, Ed., 2012, pp. 135-162.
- 8] D. K. Prasad y M. K. H. Leung, «An Ellipse Detection Method for Real Images,» *Proceedings of the 25th International Conference on Image and Vision Computing New Zealand (IVCNZ)*, pp. 1-8, 2010.
- 9] Y.-F. Liu, J.-M. Guo y C.-H. Chang, «Low Resolution Pedestrian Detection using Light Robust Features and Hierarchical System,» *Pattern Recognition*, vol. 47, nº 4, pp. 1616-1625, 2014.

A. J. Sacher y J. A. Marchant, «Fusing 3D information for crop/weeds classification,»
10] *Pattern Recognition. Proceedigns. 15th International Conference*, vol. 4, pp. 295-298,
2000.

C. Sun, M. Berman, D. Coward y B. Osborne, «Thickness measurement and crease
11] detection of wheat grains using stereo vision.,» *Pattern Recognition Letters*, nº 28(12), pp.
1501-1508, 2000.

R. Quevedo y J. M. Aguilera, «Computer Vision and Stereoscopy for Estimating
12] Firmness in the Salmon (Salmon salar) Fillets,» *Food and Bioprocess Technology*, nº 3(4),
pp. 561-567.

D. J. Sampson, Y. K. Chang, H. P. V. Rupisinghe y Q. U. Zaman, «A dual-view
13] computer-vision system for volume and image texture analysis in multiple apple slices
drying,» *Journal of Food Enginnering*, nº 127(0), pp. 49-57, 2014.

C. Ricolfe Viala y A. J. Sanchez Salmerón, «Camera calibration under optimal
14] conditions,» *Optics Express*, vol. 11, nº 19, pp. 10769-10775, 2011.

MathWorks Inc., «http://www.mathworks.es/company/?s_tid=gn_co,» 09 08 2011.
15] [En línea]. Available: <http://www.mathworks.es/es/help/images/ref/rgb2gray.html>.
[Último acceso: 12 07 2014].

Computer Vision and Image Understanding.
16]

B. Wu y R. Nevatia, «Detection and Tracking of Multiple, Partially Occluded Humans
17] by Bayesian,» *International Journal of Computer Vision*, vol. 75(2), pp. 247-266, 2007.