

MÁSTER EN INGENIERÍA DE SISTEMAS Y CONTROL

Clasificación de langostinos mediante visión artificial



Autor: Alejandro Rodríguez Rodríguez

Directoras: María Guijarro Mata-García y Matilde Santos Peñas

Septiembre 2018

Página en blanco intencionadamente

MÁSTER EN INGENIERÍA DE SISTEMAS Y CONTROL

Clasificación de langostinos mediante visión artificial

Autor: Alejandro Rodríguez Rodríguez

Directoras: María Guijarro Mata-García y Matilde Santos Peñas

Autorización

Autorizo a la Universidad Complutense de Madrid (UCM) y a la Universidad Nacional de Educación a Distancia (UNED) a difundir y utilizar con fines académicos, **no comerciales** y mencionando expresamente a sus autores, tanto la memoria de este Trabajo Fin de Máster, como el código, la documentación y/o el prototipo desarrollado.

Firmado:



Firma del alumno

RESUMEN

El presente proyecto presenta un novedoso método de clasificación del langostino, a través de la visión artificial.

Para este objetivo se ha desarrollado un sistema consistente en una primera fase de umbralización del espécimen a través de valores umbral dinámicos y mapas de probabilidad de pertenencia o no al conjunto langostino. La segunda fase consiste en la extracción de un vector de características de la imagen que sea apropiado y eficiente para poder definir cada muestra. Por último, se han empleado diferentes clasificadores con los vectores de características obtenidos a fin de encontrar aquel cuyo error sea mínimo.

Debido a la naturaleza del documento desarrollado, donde se busca un conjunto de vector y clasificador óptimo en el campo de la clasificación del langostino, que aún no se ha explotado con anterioridad, se proponen diferentes vectores de características para cada clasificador.

En el proyecto se emplea una base de datos de fotografías de cinco especies diferentes de langostino tomadas en las mismas condiciones de distancia focal y resolución. En total se dispondrá de cincuenta imágenes para cada tipo.

Para la validación y obtención de resultados se han empleado otras cincuenta imágenes tomadas en iguales condiciones, de otras muestras pertenecientes a las mismas especies a fin de comprobar el correcto desarrollo del programa creado, en función de los diferentes vectores de características en conjunto con los clasificadores propuestos.

Palabras clave: visión artificial, langostino, clasificación, vector de características, umbralización

ABSTRACT

The Project shows a new classification method for prawns, based on computer vision.

To reach this objective the system developed has been divided in three different stages. First stage is the discretization of the animal using dynamic umbral values and probability of membership to prawn set maps. Second stage is where the features of the specimen are extracted as an appropriated vector to define each sample. Third and last stage is the application of different classificatory methods with the features vector to find the best option with the minimal error.

Due to the nature of the developed document, where a couple of optimum feature vector and classificatory method to classify the prawn is the goal, which has not been deeply worked nowadays, different couples are tested.

A data base of photos is used in the project, all these pictures have been taken at similar external conditions, focal distance and resolution, from five different prawn species. The number of images is fifty per type.

To obtain and validate the results another fifty pictures from these five species have been used, also taken at similar conditions to check the correct functionality of the developed program, according to de different feature vectors and proposed classification methods.

Keywords: computer vision, prawn, classification, feature vector, discretization

ÍNDICE

1. INTRODUCCIÓN	1
1.1. MOTIVACIÓN Y OBJETIVOS	1
1.2. ESTRUCTURA DE LA MEMORIA	1
2. ESTADO DEL ARTE	3
2.1. OBTENCIÓN DEL LANGOSTINO	3
2.1.1. PESCA TRADICIONAL	3
2.1.2. PISCIFACTORÍAS	5
2.2. CLASIFICADORAS DE LANGOSTINO	7
2.2.1. A BORDO	7
2.2.2. EN TIERRA	8
2.3. VISIÓN POR COMPUTADOR	9
2.3.1. FUNDAMENTOS	9
2.3.2. EQUIPOS DE VIDEO	10
2.3.3. CARACTERÍSTICAS DE LA IMAGEN	12
2.3.4. ANÁLISIS DE LA IMAGEN	17
3. CLASIFICACIÓN DE LANGOSTINOS	27
3.1. MATERIALES: ESPECIES DE LANGOSTINO	27
3.2. IDENTIFICACIÓN DEL LANGOSTINO	29
3.2.1. PRIMERA FASE: SEGMENTACIÓN	30
3.2.2. SEGUNDA FASE: EXTRACCIÓN DE CARACTERÍSTICAS	43
3.2.3. TERCERA FASE: CLASIFICACIÓN	49
4. ANÁLISIS Y DISCUSIÓN DE RESULTADOS	69
5. CONCLUSIONES Y FUTUROS TRABAJOS	73
6. BIBLIOGRAFÍA	75
7. ANEXOS	77
7.1. BASE DE DATOS	77
7.2. IMÁGENES PROCESADAS	82
7.3. RESULTADOS	83
7.3.1. VECTOR $x_{Oi} = Ray, t, R, G, B$	83
7.3.2. VECTOR $x_{Ai} = ColorA, t, rc$	86
7.3.3. VECTOR $x_{Bi} = ColorB, t, rc$	89
7.4. FUNCIÓN DE CLASIFICACIÓN	92

Página en blanco intencionadamente

LISTA DE FIGURAS

Figura 1. Flujograma del proyecto	1
Figura 2. Sistema propuesto	2
Figura 3. Tangonero [3].....	3
Figura 4. Maniobra de tangón	4
Figura 5. Parque de pesca, buque Giuseppe.....	4
Figura 6. Piscifactoría [8].....	5
Figura 7. Proceso de cría [3].....	6
Figura 8. Parque de pesca buque Alver.	7
Figura 9. Clasificadora mecánica [5]	8
Figura 10. Ciudad del cabo BN	9
Figura 11. Ciudad del cabo color.....	10
Figura 12. CCD izqda. CMOS dcha. [7]	11
Figura 13. Espacio RGB.....	13
Figura 14. Motor y bomba RGB.....	14
Figura 15. Espacio HSI [5].....	15
Figura 16. Motor y bomba HSI	15
Figura 17. Motor y bomba XYZ	16
Figura 18. Convolución [5]	17
Figura 19. Histograma buque Ribadavia BN	18
Figura 20. Histograma buque Ribadavia color	18
Figura 21. Interpolación bilineal [5]	20
Figura 22. Interpolación bicúbica [5]	20
Figura 23. Correlación [5].....	21
Figura 24. Tipos habituales de ruido [5]	22
Figura 25. Método de Canny [10]	24
Figura 26. Penaeus Kerathurus	27
Figura 27. Penaeus Monodon.....	27
Figura 28. Penaeus Peche	28
Figura 29. Penaeus Vannamei.....	28
Figura 30. Jumbo.....	28
Figura 31. Esquema del proceso	29
Figura 32. Esquema trabajo primera fase.....	30
Figura 33. Imagen escala de grises	31
Figura 34. Umbralización con diferentes valores fondo blanco	32
Figura 35. Umbralización con diferentes valores fondo coloreado	32
Figura 36. Umbralización con Otsu	33
Figura 37. Umbralización simple fondo metálico	34
Figura 38. Umbralización por color fondo metálico	34
Figura 39. Umbralización por color fondo coloreado	35
Figura 40. Umbralización por color fondo metálico 2	35
Figura 41. Comparativa umbralizaciones.....	37
Figura 42. Langostino umbralizado con objetos indeseados.....	39
Figura 43. Mapas de pertenencia	40

Figura 44. Mapa de pertenencia definido.....	41
Figura 45. Imagen discretizada para diferentes límites.....	42
Figura 46. Pasos realizados por el programa en discretización[Elaboración propia].....	42
Figura 47. Extracción de bordes.....	43
Figura 48. Esquema trabajo segunda fase.....	43
Figura 49. Imagen procesada.....	44
Figura 50. Comparativa extracción de rayado.....	47
Figura 51. Esquema trabajo tercera fase.....	50
Figura 52. Distribución bidimensional rayado-tamaño.....	52
Figura 53. Distribución bidimensional exp(rayado)-tamaño.....	52
Figura 54. Representación 3D vector de características A.....	54
Figura 55. Representación 2D gris-exp(rayado) vector de características A.....	54
Figura 56. Representación 2D gris-tamaño vector de características A.....	55
Figura 57. Representación 2D R-G.....	56
Figura 58. Representación 2D R-B.....	57
Figura 59. Representación 2D G-B.....	57
Figura 60. Representación 2D Rayado-R.....	58
Figura 61. Representación 2D Rayado-G.....	59
Figura 62. Representación 2D Rayado-R/G.....	59
Figura 63. Representación 2D Rayado-R/B.....	60
Figura 64. Representación 2D Rayado-(R-B).....	60
Figura 65. Representación 2D Rayado-(R-G).....	61
Figura 66. Representación 3D vector de características B.....	62
Figura 67. Representación 2D color-exp(rayado) vector de características B.....	63
Figura 68. Representación 2D color-tamaño vector de características B.....	63
Figura 69. Diagramas de Voronoi bidimensionales.....	65
Figura 70. Representación bidimensional método k-NN.....	66

LISTA DE TABLAS

Tabla 1. Tamaño medio por especie BBDD.....	51
Tabla 2. Color medio por especie BBDD.....	53
Tabla 3. Porcentaje de acierto según método, vector y tipo de langostino.....	70
Tabla 4. Porcentaje de acierto según método y vector para el conjunto de langostinos.....	71

1. INTRODUCCIÓN

1.1. MOTIVACIÓN Y OBJETIVOS

Actualmente se emplean sistemas tradicionales y manuales en la clasificación de langostinos tanto a bordo de los buques como en las factorías, lo que supone un atraso tecnológico frente a otros sectores donde la visión artificial ya es una realidad en la clasificación de diferentes elementos sobre el entorno.

El objetivo del proyecto es el desarrollo de un sistema de clasificación de langostinos mediante visión artificial que pueda ser aplicado tanto en factorías como a bordo, si se diseña un sistema físico apropiado y acorde.

Parte del objetivo es lograr una tasa de acierto tan elevada como sea posible, ya que es un proyecto destinado a la manipulación de alimentos para consumo humano, donde los estándares de calidad deben ser lo más elevados. Por ello, aun aplicando este sistema, debería ser requisito en su hipotético uso, la supervisión de un experto humano para asegurar que dicha calidad se mantiene.

1.2. ESTRUCTURA DE LA MEMORIA

El sistema de clasificación del langostino por visión artificial se realiza a través de software y hardware. El presente proyecto se centra en el desarrollo a través de software. Sin embargo, es conveniente mencionar algunos detalles a nivel hardware a fin de justificar la posible implantación de dicho sistema.

En el proyecto se va a entender que los langostinos son colocados en una cinta, donde un equipo de visión toma una imagen del espécimen, y es enviada a un ordenador para procesarla, siguiendo una serie de fases (figura 1), en las que se va a dividir la memoria.

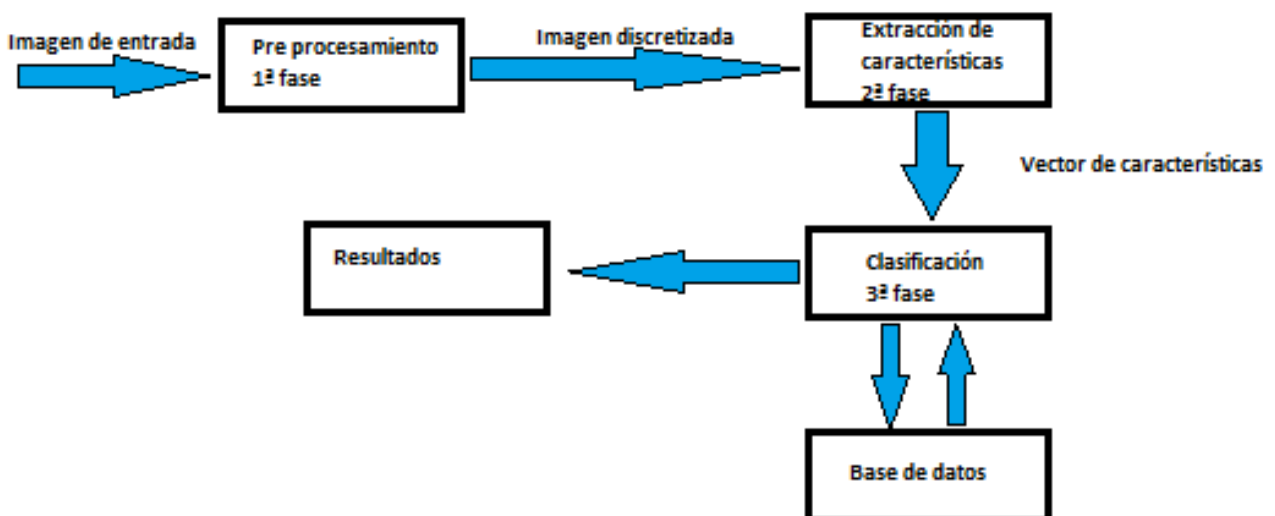


Figura 1. Flujograma del proyecto

Una cámara de video tomará las imágenes de los especímenes transportados en la cinta, tras lo cual procesará la imagen, clasificará la muestra por tipo y tamaño y, mediante un sistema mecánico, los irá derivando a su destino correctamente.

La imagen se toma con una cámara digital Canon 4000D a una distancia focal de 750 mm, en el presente proyecto, comprimida en formato JPEG con dimensiones 600x400 para reducir coste computacional, ya que no afecta apenas al resultado a esos niveles.

Como se puede ver en la figura 2, se propone un sistema donde los langostinos son, una vez fotografiados por la cámara y procesados por el ordenador, distribuidos por las diferentes rampas con un simple sistema de paletas rotadas en el eje Z a partir de un sistema apropiado, que puede ser eléctrico o hidráulico.

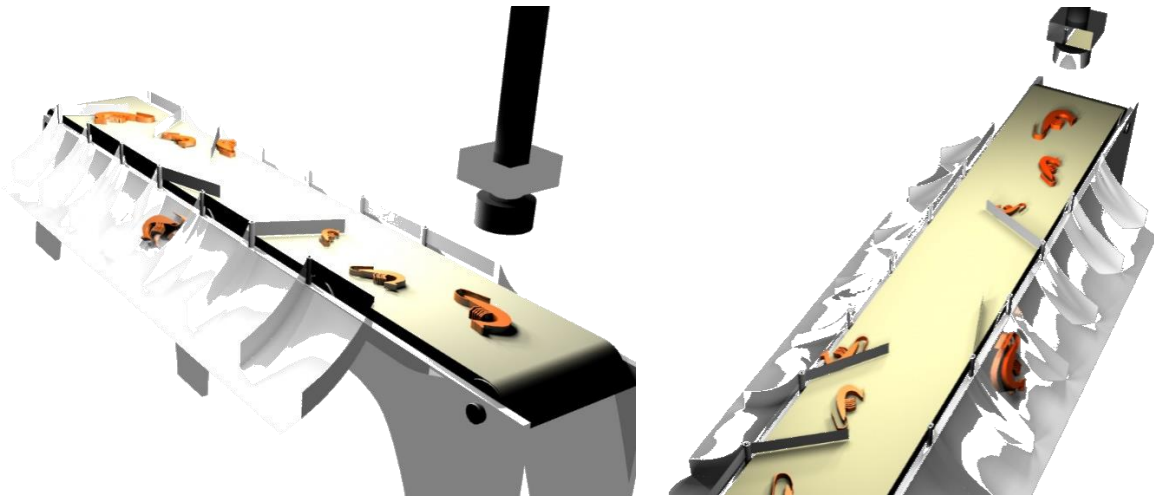


Figura 2. Sistema propuesto

2. ESTADO DEL ARTE

En este apartado se realiza un breve resumen del estado del arte en la captura de langostino en términos globales, así como de las clasificaciones de la especie a bordo o en tierra. En cuanto a la visión artificial se tratan y presentan algunas de las técnicas empleadas actualmente.

2.1. OBTENCIÓN DEL LANGOSTINO

2.1.1. PESCA TRADICIONAL

La pesca en el mar ha acompañado a la humanidad prácticamente desde sus comienzos, evolucionando a la par que lo hacía el ser humano y sus habilidades.

En nuestros días la pesca de langostino se realiza con un tipo de embarcación especial llamada buque de pesca de arrastre con tangón, o tangonero. La pesca de este tipo de buques es, en esencia, similar a la pesca de arrastre tradicional, donde una red es largada por la popa del buque hasta la profundidad apropiada. Una vez se llena la misma con la especie buscada, mediante unas maquinillas consistentes en motores conectados a rodillos se vira o recoge el cable hasta subir la red a bordo.

La diferencia fundamental, consiste en que los buques tangoneros no largan la red por la popa del buque, donde una rampa ayuda a recogerla, sino que se larga por los costados, con ayuda de unos palos llamados tangones, como se ve en la figura 3.



Figura 3. Tangonero [3]

Al no faenar en aguas tan profundas estos palos pueden soportar la fuerza ejercida por la red durante el arrastre y virada.

Durante las maniobras de pesca los tangones se abren como alas permitiendo la operación que se muestra en el esquema de la figura 4.

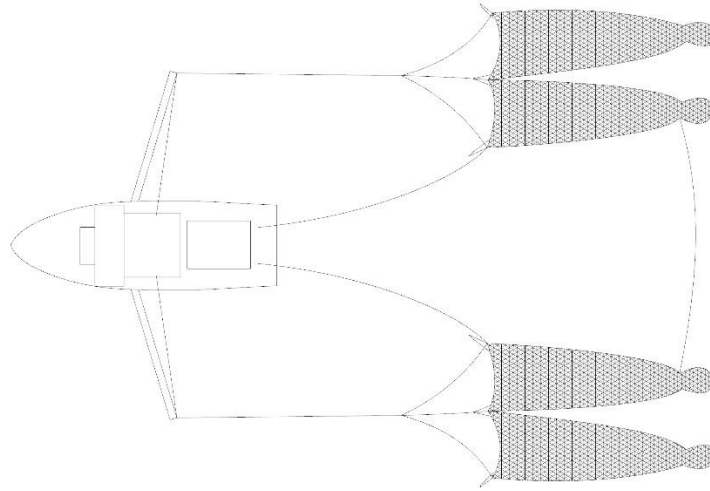


Figura 4. Maniobra de tangón

Con esta disposición se consiguen cuatro sacos de langostino, maximizando así la pesca de forma eficiente. Una vez estos sacos son recogidos y descargados a bordo en la bandeja de recepción (figura 5), primero se lavan y luego se colocan sobre una mesa de clasificación, lo que se realiza a mano, por especie y tamaño, colocándolas en cajas que posteriormente se congelan en unos túneles a bordo y almacenan en la bodega.



Figura 5. Parque de pesca, buque Giuseppe

2.1.2. PISCIFACTORÍAS

Además del método tradicional de captura, que actualmente sigue teniendo un gran potencial para capturar langostino salvaje, existen criaderos o piscifactorías donde se cultivan estos animales (figura 6), si bien son de menor calidad que el salvaje. Este tipo de piscifactoría se inició a comienzos del siglo XX, aunque no alcanzó su desarrollo comercial y a gran escala hasta el tercer cuarto de dicho siglo. En la actualidad diferentes compañías invierten por todo el mundo en esta clase de criaderos debido a su menor coste de explotación que la captura del langostino salvaje [1].

Este sistema de obtención del animal tiene una duración aproximada de un año, contando desde la fecundación en los llamados tanques de cortejo, donde machos y hembras se mantienen alimentados y controlados durante cuatro meses, hasta la cosecha final tras aproximadamente otros cinco meses en los que han estado en grandes piscinas engordando (figura 7).



Figura 6. Piscifactoría [8]

La ventaja de este sistema es sin duda el abaratamiento de costes frente a la pesca tradicional, ya que no se precisa emplear buques con su tripulación, y permiten un mayor control del producto mediante análisis continuos por estar la población en un ambiente controlado. Este abaratamiento frente al producto salvaje se viene notando en el incremento de piscifactorías para el cultivo del langostino en las últimas décadas.

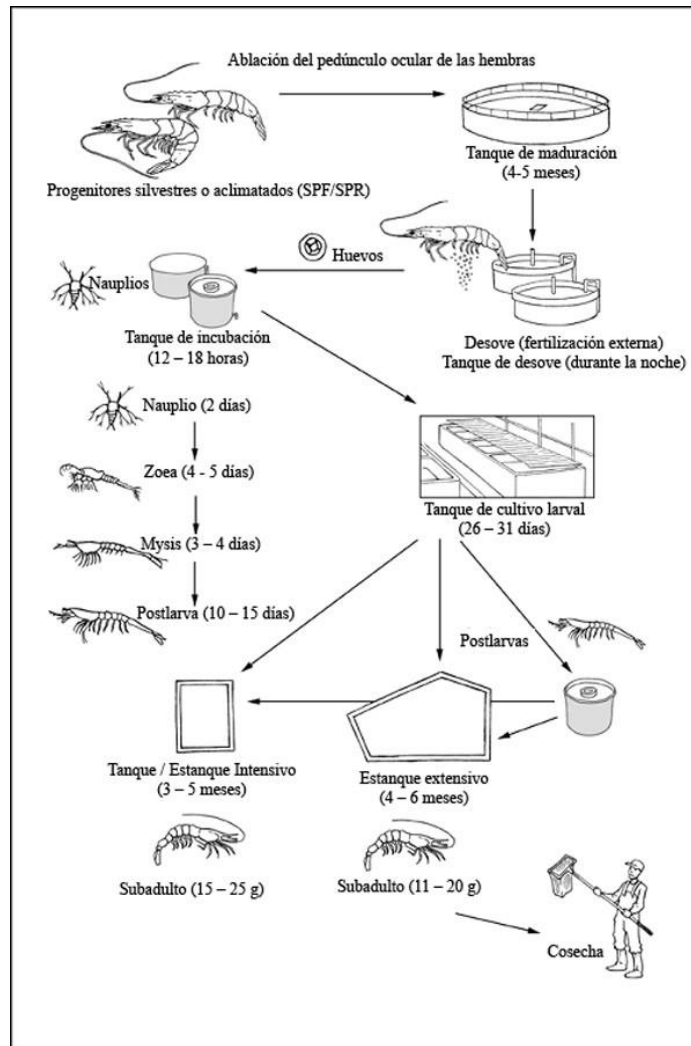


Figura 7. Proceso de cría [3]

La principal desventaja de este método es la propia calidad del producto, ya que el langostino de cultivo no tiene las propiedades del salvaje en relación al sabor y la textura, conllevando por tanto una reducción del precio de producto.

2.2. CLASIFICADORAS DE LANGOSTINO

2.2.1. A BORDO

En la actualidad, el sistema de clasificación de langostinos sigue siendo manual, sobre todo a bordo de los buques. Esto es, una vez el langostino es descargado en el pantano o mesa de recepción, se traslada a unas cubas donde se lavan y colocan sobre unas mesas grandes para su clasificación (figura 8).



Figura 8. Parque de pesca buque Alver.

Alrededor de estas mesas, un número variable de personas se dedican visual y manualmente a seleccionar y coger los especímenes de acuerdo con su tipo y tamaño, para irlos colocando en cajas de forma ordenada.

Este sistema tradicional tiene la ventaja de que personas con mucha experiencia pueden realizar el trabajo visualmente, a gran velocidad y sin mucho error. Sin embargo, para llegar a esos niveles de conocimiento y experiencia, estas personas requieren invertir mucho tiempo en este proceso. Terminado ese empacamiento se pesan las cajas para ver que están en los valores correspondientes y se almacenan en una bodega refrigerada.

2.2.2. EN TIERRA

Existen por otro lado, las clasificadoras en tierra de las factorías, usadas sobre todo en sitios donde las mareas (tiempo de pesca entre salida y llegada a puerto de los buques) son de menos de una semana, normalmente uno o dos días, de tal forma que lo único que se hace a bordo es separar el langostino de posibles especies cogidas accidentalmente, como diversos tipos de pescado.

Estas clasificadoras pueden ser de varios tipos diferentes, todos ellos mecánicos, siendo el más común la clasificadora de rodillos, donde los langostinos pasan por una serie de rodillos entre los que la separación varía, permitiendo que vayan cayendo en diferentes tolvas según sus dimensiones.

En cualquier caso, no existen clasificadoras que segreguen por especie, únicamente por tamaño (figura 9).



Figura 9. Clasificadora mecánica [5]

Como se ha mencionado en el párrafo anterior, pueden existir variaciones de clasificadoras mecánicas: mediante paletas, ranuras en las cintas, u otros métodos. En general, existe la posibilidad de que algunos especímenes no entren en el lugar que les corresponde, por lo que siempre hay supervisión humana en el proceso post clasificadora.

También es común encontrar en factorías de tierra mesas de clasificación manual como las instaladas a bordo, en algunos casos por preferencia del responsable, al entender que no son suficientemente eficientes las clasificadoras existentes en el mercado.

2.3. VISIÓN POR COMPUTADOR

Se presenta en este apartado una idea general de los fundamentos, equipos involucrados y diferentes métodos o técnicas de análisis empleados en la visión por computador.

2.3.1. FUNDAMENTOS

La visión por computador es, en términos globales, la representación numérica o analítica del mundo que nos rodea en lo referente a imágenes, es decir, una representación de las formas y los colores a través de resultados numéricos: la entrada al sistema es una imagen, y la salida es información computacional.

Toda imagen puede representarse como una matriz, de $m \times n$ píxeles, adoptando cada píxel un valor. En casos de imágenes monocromáticas, o en escala de grises, normalmente entre 0 y 255 que corresponde con un valor en la escala de grises, donde el 0 es el negro, el 255 el blanco y cualquier punto intermedio el nivel de gris correspondiente.

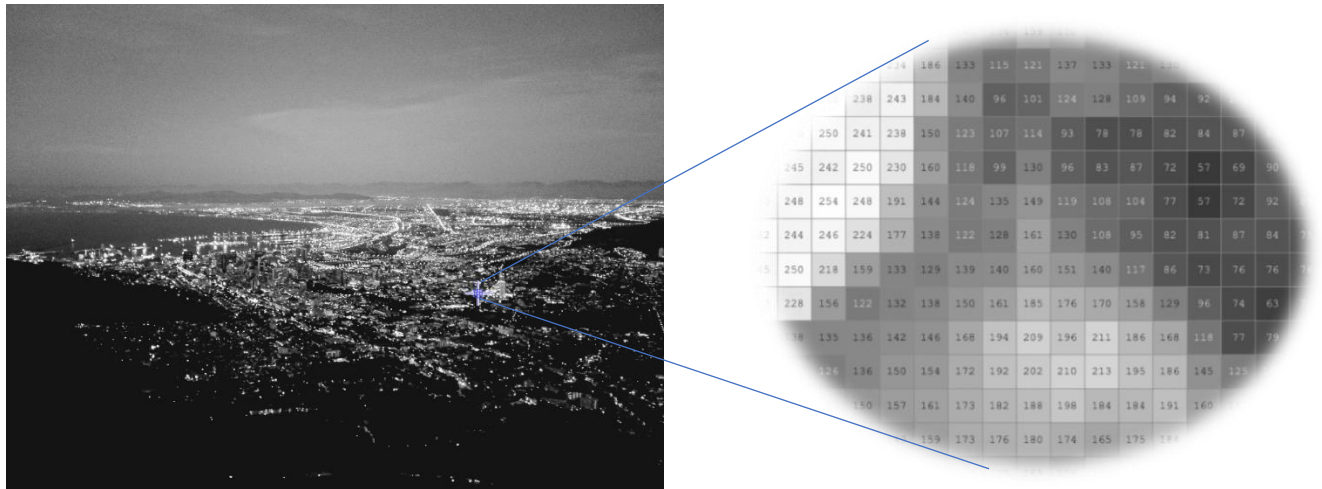


Figura 10. Ciudad del cabo BN

En imágenes monocromáticas binarias, valores de 0 representarán al negro y valores de 1 al blanco.

Se puede observar en la figura 10 que cada píxel en la imagen mostrada tiene un valor único entre 0 y 255.

En imágenes a color, la información que recibe el ordenador es, de forma general, un grupo de tres matrices de $m \times n$ píxeles, donde cada una tiene un valor de 0 a 255, para cada posición, y cada color entre rojo, verde y azul (RGB, por sus siglas en inglés). De esta forma, donde por ejemplo predomine el rojo, la matriz R tendrá en esos puntos valores más cercanos al 255; donde predomine el magenta, se encontrarán valores más elevados y similares en la matriz R y B, mientras que la matriz G tendrá valores más reducidos en dichos píxeles (figura 11).

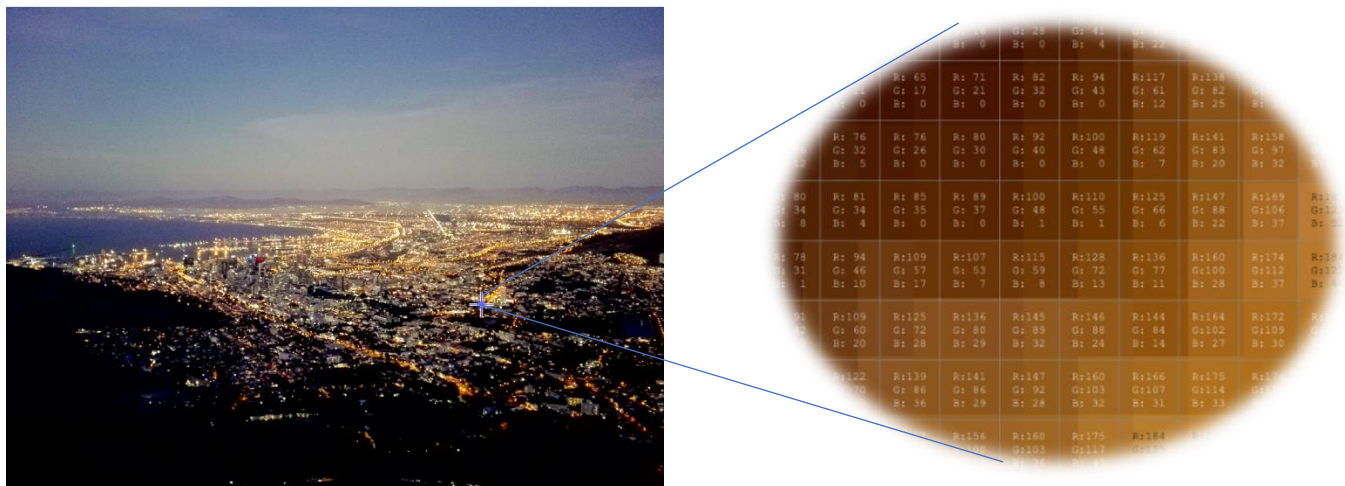


Figura 11. Ciudad del cabo color

Existen otros espacios de color para trabajar con las imágenes, si bien el presentado es el más común e intuitivo para tener una idea general de cómo es trasladada la imagen a información procesable por el ordenador.

En el punto correspondiente al análisis de las imágenes se hablará de dichos espacios de color para conocer las posibles alternativas y relación que existe entre ellas.

2.3.2. EQUIPOS DE VIDEO

Para trabajar con imágenes del mundo que nos rodea se necesitan dos elementos fundamentales, el ordenador para procesar la información, y la cámara para obtenerlas.

Si se habla de las cámaras, y tomando como referencia lo expuesto en [5], se distingue en primera instancia dos grupos diferentes en función del sensor que incorporen: dispositivo de carga acoplada (CCD, por sus siglas en inglés) o semiconductor complementario de óxido metálico (CMOS, por sus siglas en inglés). Los primeros se basan en elementos sensibles a la luz que, al ser activados por un reloj, crean un pozo de potencial. Al llegar la luz se producen electrones que quedan atrapados en este pozo de potencial, siendo su número proporcional a la intensidad luminosa que reciben. Usando registros de desplazamiento, se lee la imagen línea a línea, utilizando convertidores analógico-digitales. De esta forma, se consigue que el valor de intensidad lumínica de cada punto del sensor se corresponda con cada píxel de la imagen.

Las cámaras CMOS son más modernas que las cámaras CCD. En estos casos, la conversión de luz da un valor digital mediante el uso de varios transistores en el propio elemento sensible. Por lo tanto, a diferencia de las cámaras CCD, no necesitan registrar este desplazamiento ni disponer de un convertor analógico-digital.

Para elegir una u otra opción hay que analizar primero el problema que se va a tratar, teniendo en cuenta las ventajas y desventajas que puede ofrecer cada una.

La primera ventaja que encontramos, al comparar las cámaras de tipo CCD y CMOS, es que estas últimas realizan la conversión analógico-digital en el mismo fotodiodo, por lo que los sensores son más reducidos, ya que no son necesarios elementos electrónicos para el desplazamiento de carga.

Por otro lado, se puede acceder a cada píxel de forma individual, por lo que no se necesita leer toda la imagen de una sola vez, como ocurre con las cámaras de tipo CCD. Por ello, es frecuente que, en las cámaras de tipo CMOS, se defina lo que se conoce como una región de interés (ROI, por sus siglas en inglés), que se puede leer, sin necesidad de obtener el resto de la imagen. Se consiguen así velocidades de transferencia superiores, al centrarnos en una región más pequeña de la imagen, que es sobre la que se trabaja.

Al ser la tecnología de CMOS igual a la de otros dispositivos electrónicos, como memorias o microprocesadores, el coste de fabricación es menor, y se pueden integrar en circuitos electrónicos de control dentro del mismo dispositivo de captura, lo cual influirá evidentemente en el proyecto final a la hora de analizar los costes.

Otro punto a destacar como ventaja es que el consumo eléctrico de las cámaras de tipo CMOS es más reducido que el de las cámaras de tipo CCD.

Por último, e importante también, las cámaras de tipo CMOS no presentan efecto *blooming*, que sí sufren las cámaras de tipo CCD. Este efecto se da cuando un píxel de una cámara CCD recibe mucha luz, por lo que el número de electrones que desprende el elemento sensor es mucho mayor del que es capaz de almacenar, desbordando y transmitiendo de esta manera la carga que le sobra a los píxeles vecinos, llegándose a saturar (alcanzar valores máximos) y, por lo tanto, propagando el efecto a lo largo del sensor. Por esta razón las cámaras CCD son más sensibles a focos de luz potentes (figura 12).



Figura 12. CCD izqda. CMOS dcha. [7]

En resumen, las cámaras CMOS tienen la ventaja de mejor acceso, más rápido y eficiente, un consumo menor y no sufren el efecto blooming.

Por otro lado, la ventaja de las cámaras CCD es la calidad de la imagen, que es mayor. Presentan mayor sensibilidad al ser mejor la relación señal a ruido (esto se debe a que se tienen procesadores de señal externos al propio sensor, lo que permite añadir sistemas más complejos y sofisticados). Además, dispone de menos píxeles defectuosos y la ganancia de cada uno de ellos es uniforme con un número de píxeles de los sensores más elevado.

2.3.3. CARACTERÍSTICAS DE LA IMAGEN

Ya se ha hablado en el primer apartado del presente capítulo de cómo una imagen es obtenida y convertida en información procesable por un ordenador, es decir, convertida en una matriz o grupo de matrices numéricas.

Cada punto bidimensional de estas matrices de $m \times n$: $f(x,y)$ representa un píxel, y el valor de éste la intensidad o nivel de gris del mismo, siendo para el caso habitual de 256 niveles de gris 0, un objeto sin iluminación ninguna y 255, un objeto que refleja todo rayo incidente sobre él.

$$f(x,y) = \begin{bmatrix} f(0,0) & f(1,0) & \dots & f(N-2,0) & f(N-1,0) \\ f(0,1) & f(1,1) & \dots & f(N-2,1) & f(N-1,1) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ f(0,M-2) & f(1,M-2) & \dots & f(N-2,M-2) & f(N-1,M-2) \\ f(0,M-1) & f(1,M-1) & \dots & f(N-2,M-1) & f(N-1,M-1) \end{bmatrix}$$

El número 256 se refiere al caso habitual en que cada píxel ocupa un byte, es decir, 8 bits, y el nivel de grises viene dado por $2^{n^{\text{bits}}}$, que resulta $2^8=256$.

Cabe destacar en este punto un concepto importante en las imágenes. Además de las vecindades de píxeles en la matriz, resulta muy común trabajar con distancias entre píxeles. Si bien existen diversos métodos, el más común por sencillez y efectividad es la distancia euclídea (1) entre los puntos $p=(s,t)$ y $q=(x,y)$:

$$D_E(p,q) = \sqrt{(x-s)^2 + (y-t)^2} \quad (1)$$

De esta forma valga como ejemplo para ayudar en la comprensión, un subespacio 5x5 de la matriz de $m \times n$, centrado en el (3,3) de dicho subespacio, las distancias serían:

$$\begin{matrix} \sqrt{8} & \sqrt{5} & 2 & \sqrt{5} & \sqrt{8} \\ \sqrt{5} & \sqrt{2} & 1 & \sqrt{2} & \sqrt{5} \\ 2 & 1 & 0 & 1 & 2 \\ \sqrt{5} & \sqrt{2} & 1 & \sqrt{2} & \sqrt{5} \\ \sqrt{8} & \sqrt{5} & 2 & \sqrt{5} & \sqrt{8} \end{matrix}$$

Otras distancias pueden ser más rápidas en su cálculo, pero menos exactas, y debe valorarse en función del proyecto a desarrollar.

Con esto se tendría definida la imagen monocromática o en niveles de gris. Sin embargo, lo habitual es trabajar, en la mayoría de los casos con imágenes a color, mucho más completas y con una mayor información. Para trabajar con imágenes así ya no es suficiente hablar del nivel de gris, valor monocromático, si no que se debe definir el espacio de color en que se va a trabajar.

Los espacios de color normalmente tienen tres componentes por píxel para poder definirlo correctamente, siendo esta la razón por la que, los resultados son un vector de tres matrices, una por componente.

Los principales espacios de color se definen a continuación [5].

Espacio de color RGB

Consistente en dar a cada píxel un valor para sus componentes roja (R = red), verde (G = green) y azul (B = blue), y el resultado se obtiene mediante la suma de las tres componentes: $X=R+G+B$.

La representación gráfica de este espacio es un cubo (figura 13), siendo cada uno de sus ejes una de estas componentes. De esta forma los puntos intermedios (con parte de dos o más de estos) supondrán el espectro de colores completo.

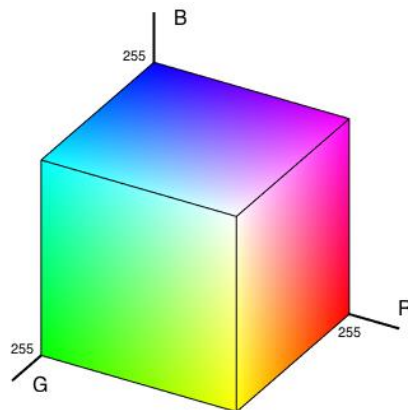


Figura 13. Espacio RGB

Este sistema es el más intuitivo y uno de los más empleados. Sin embargo, presenta el inconveniente de que, si bien la información de color la adquiere sin error, no es capaz de diferenciar de forma correcta, o separadamente del color, la información referente a intensidad, lo que en según qué casos puede dificultar el trabajar con este espacio.

En la figura 14 pueden verse la imagen original RGB (14.a), y las componentes R, G y B (14.b, 14.c y 14.d respectivamente).

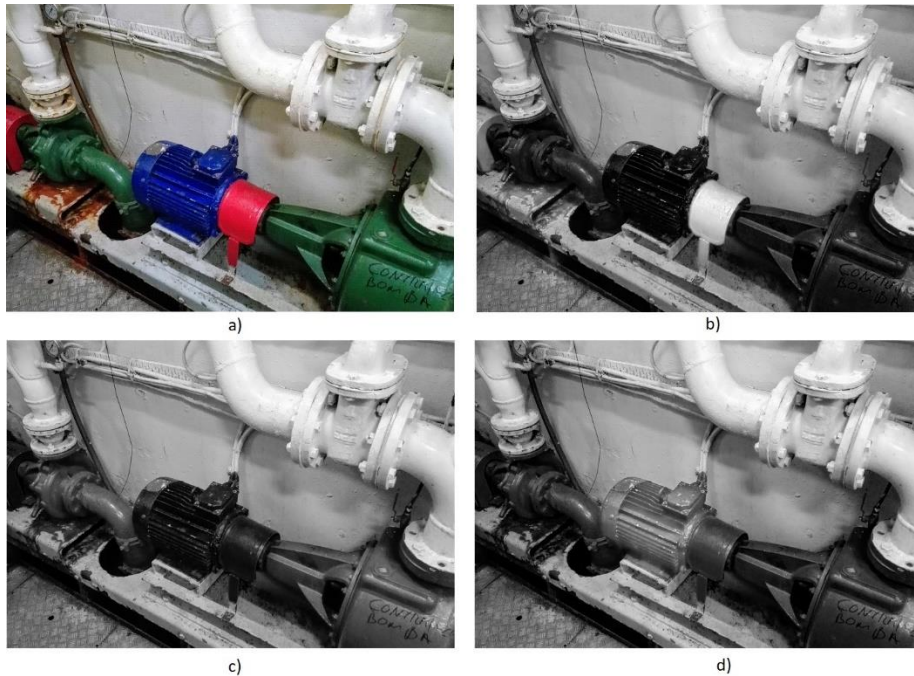


Figura 14. Motor y bomba RGB

Espacio HSI

En este espacio sus componentes se refieren a características de la imagen, en concreto:

- H: es el tono (*hue* en inglés), que podría entenderse como la longitud de onda del color
- S: es la saturación (*saturation* en inglés), que se entiende como la relación entre color y brillo
- I: es la intensidad (*intensity* en inglés) o brillo

Así, de dos fuentes de luz con mismo espectro, aquella que tenga mayor intensidad aparecerá más brillante. Existen sin embargo ejemplos de objetos de igual intensidad que no tienen el mismo brillo: una luz verde y otra roja se distinguen por el tono que representa la longitud de onda de ese color. Sin embargo, no es igual a la longitud de onda ya que algunos colores que se encuentran en la naturaleza (el púrpura, por ejemplo) no se encuentran en la descomposición de la luz blanca. La saturación distingue la blancura de un color (el rojo tiene saturación máxima y el rosa mínima) [5].

La conversión o paso entre el espacio HSI y el RGB es relativamente sencilla, al venir definida por las siguientes expresiones analíticas (2):

$$\begin{aligned}
 I &= \frac{R + G + B}{3} \\
 H &= \arctan\left(\frac{\sqrt{3}(G - B)}{(R - G) + (R - B)}\right) \\
 S &= 1 - \frac{\min(R, G, B)}{I}
 \end{aligned}
 \tag{2}$$

Existen, sin embargo, otras transformaciones enunciadas por diversos autores con mayor o menor precisión, según el caso [5].

La representación de este espacio de color consiste en dos conos unidos por la base (figura 15). El eje vertical que iría entre negro y blanco sería, la intensidad, que indicaría a que altura se cortaría la figura. Una vez en ese corte, el tono indicaría el ángulo, siendo el rojo 0º; el verde 120º y el azul 240º. La saturación, por último, señala la distancia al centro de la base.

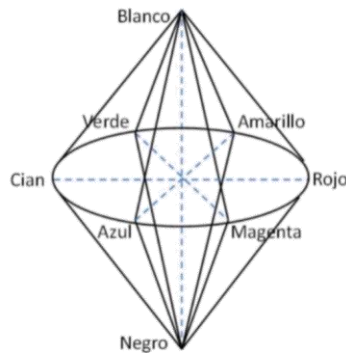


Figura 15. Espacio HSI [5]

En la figura 6ª se muestra la imagen HSI, siendo las componentes H, S e I las figuras 16.b, 16.c y 16.d respectivamente.

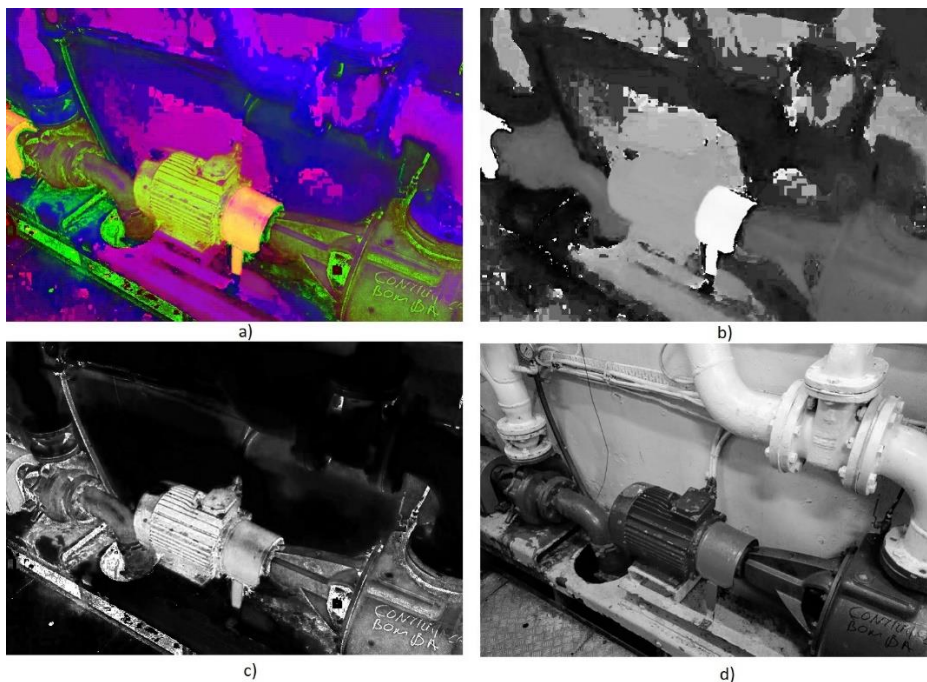


Figura 16. Motor y bomba HSI

Espacio de color XYZ

En el espacio HSI pueden aparecer coeficientes negativos, lo que para determinados cálculos específicos puede suponer una fuente de error. Por ello se definió el espacio de color XYZ (3).

$$x = \frac{X}{X+Y+Z} \quad y = \frac{Y}{X+Y+Z} \quad z = \frac{Z}{X+Y+Z} \quad (3)$$

En la componente Y se encuentra la luminosidad mientras que en XZ la coloración. Para una determinada luminosidad es (4):

$$X = \frac{xY}{y} \quad Z = \frac{zY}{y} \quad (4)$$

La conversión entre espacios de color, concretamente entre RGB y XYZ queda definida por la siguiente expresión matricial (5) de acuerdo con la ITU (International Telecommunications Union) [15]:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.431 & 0.342 & 0.178 \\ 0.222 & 0.707 & 0.071 \\ 0.020 & 0.130 & 0.939 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad \begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 3.063 & -1.393 & -0.476 \\ -0.969 & 1.876 & 0.042 \\ 0.068 & -0.229 & 1.069 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad (5)$$

Siguiendo el ejemplo de los espacios anteriores, podemos ver los resultados usando la primera de estas matrices en la figura 17.a, siendo las componentes X, Y y Z las mostradas en las figuras 17.b, 17.c y 17.d respectivamente.

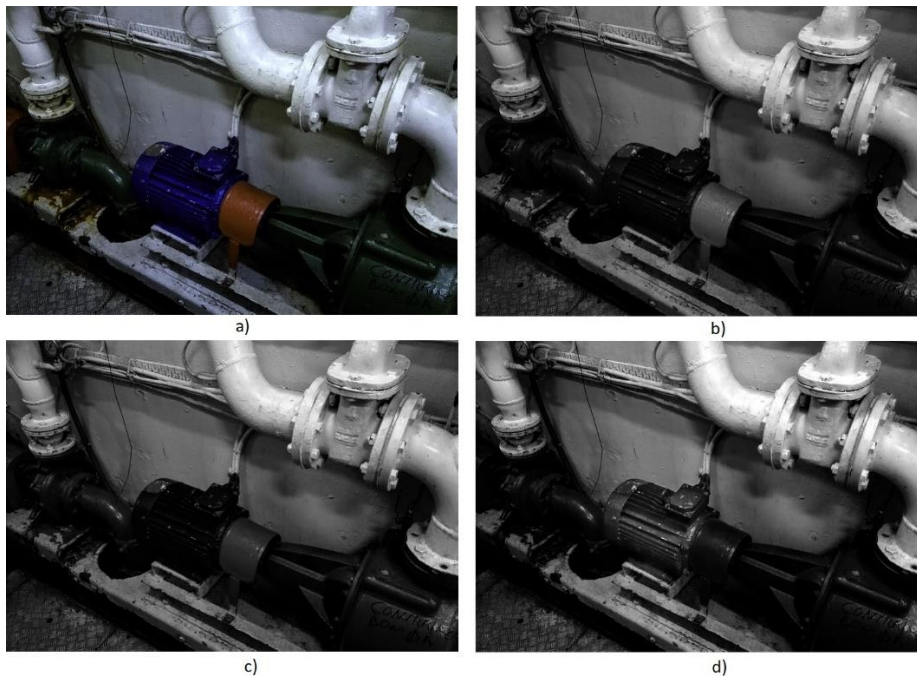


Figura 17. Motor y bomba XYZ

2.3.4. ANÁLISIS DE LA IMAGEN

Se han expuesto hasta ahora los métodos de obtención de las imágenes y su posterior conversión a valores con los que poder trabajar informática y analíticamente, mediante las descomposiciones en diferentes espacios de color, relacionados entre sí.

En este apartado se van a presentar diferentes opciones para estudiar y comprender las características que definen a una imagen, de forma que esto permita poder trabajar con ellas, identificar unos elementos, aislar otros, comparar dos matrices, etc.

Convolución

La convolución [8], es un operador matemático que transforma dos funciones f y g en una tercera función que representa la magnitud en la que se superponen f y una versión trasladada e invertida de g .

Cuando se habla de este concepto, en realidad se podría decir que es un filtro que se aplica a la imagen píxel a píxel, o más habitualmente región a región de píxeles, normalmente 3×3 o 5×5 .

Sin entrar en un desarrollo matemático que no resulta demasiado complejo en su forma analítica, se podría decir que esta región de la matriz tiene unos valores que se multiplican centrándose para cada píxel de la matriz de la imagen. El sumatorio de estos resultados conforman el píxel de la matriz imagen resultante, como se puede observar en el ejemplo de la figura 18, para una convolución de gradientes, muy útil entre otras cosas en el análisis de bordes.

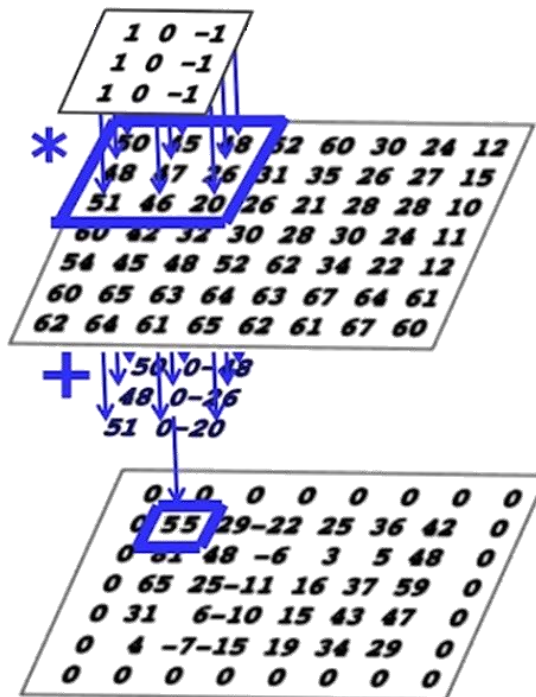


Figura 18. Convolución [5]

Histogramas

Los histogramas son una poderosa herramienta para analizar de forma visual y analítica la distribución de valores que toman los píxeles en el cómputo de la imagen. La definición de este concepto [8], es una representación gráfica de una variable en forma de barras, donde la superficie de cada barra es proporcional a la frecuencia de los valores representados.

Por ejemplo, una imagen monocromática con 256 niveles de gris tendría un histograma donde el eje horizontal indicaría el valor del nivel de gris y el vertical el número de píxeles que tienen dicho valor numérico.

En la figura 19 se puede observar la imagen de un buque con 256 niveles de gris, repartidos de forma bastante homogénea y su histograma. Hay un elevado número de píxeles que son completamente negros, y eso coincide con lo que presenta la imagen, donde en la zona derecha hay una región importante de dicho color (valor del píxel, cero). Se descubre también un alto número de píxeles cercanos al valor máximo (255), aunque no llegan a ese valor, que son los que forman la zona alta del buque, casi blancos, y su reflejo.

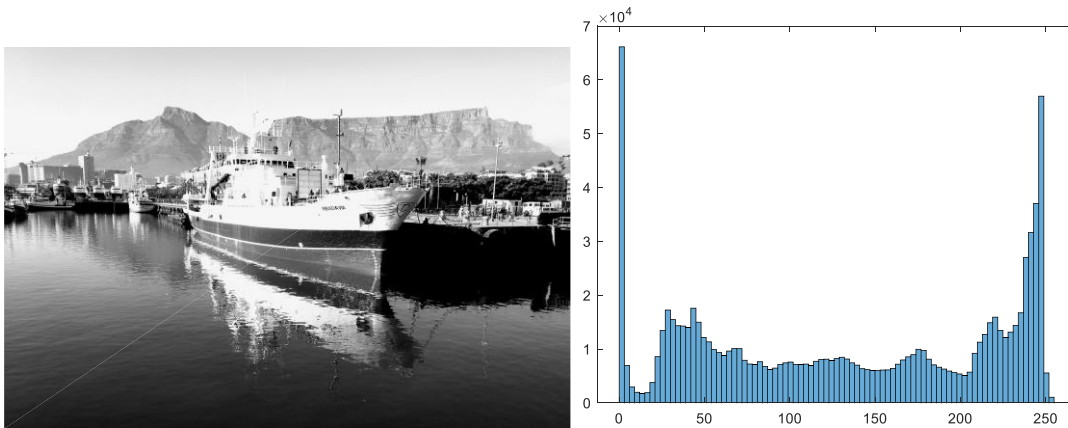


Figura 19. Histograma buque Ribadavia BN

Esta misma imagen en color tendría no uno, sino tres histogramas diferentes, uno para cada matriz de valores R, G y B, mostrados respectivamente en la figura 20, señalando la predominancia de tonalidad roja que tiene la imagen con una mayor agrupación de píxeles en la matriz R de valores elevados.

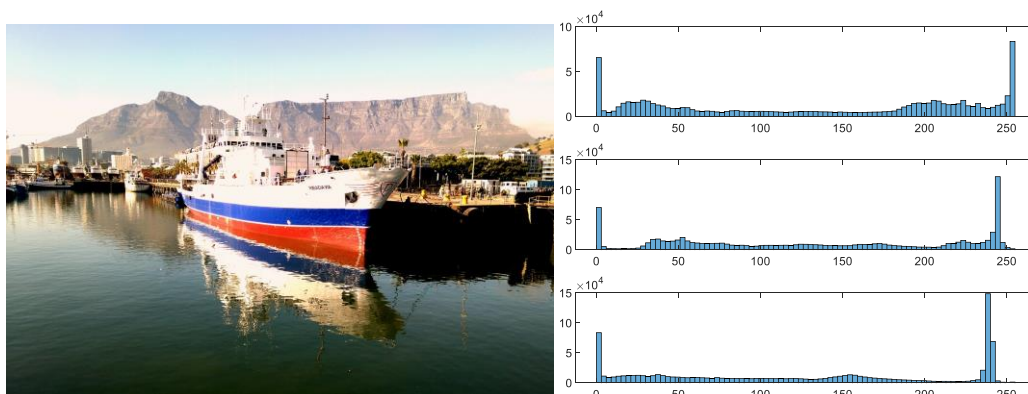


Figura 20. Histograma buque Ribadavia color

Transformaciones geométricas

Al analizar la imagen, puede ocurrir que no se encuentre en el estado óptimo por cuestiones geométricas, como puede ser un tamaño inadecuado o una posición inesperada. Para solucionar esto se presentan las transformaciones geométricas sobre la imagen obtenida.

Es importante entender que, al realizar una operación de este tipo sobre la matriz con valores enteros, el resultado no tiene por qué ser necesariamente un valor igualmente entero, por lo que habrá que buscar el valor nuevo resultante mediante un algoritmo de interpolación que determine el nivel de gris que se obtiene a partir de la imagen original.

Las siguientes transformaciones son algunas de las presentadas en [5].

- Traslación

La más simple de estas operaciones, consistente en desplazar cada píxel (subíndice i) una distancia determinada en una dirección definida (subíndice 0) (6). Si dicha dirección es cardinal y la distancia un número entero, el resultado (subíndice f) será directo; en caso contrario habrá que realizar una interpolación.

$$\begin{aligned}x_f &= x_i + x_0 \\y_f &= y_i + y_0\end{aligned}\tag{6}$$

En forma matricial (7)

$$\begin{bmatrix}x_f \\y_f \\1\end{bmatrix} = \begin{bmatrix}1 & 0 & x_0 \\0 & 1 & y_0 \\0 & 0 & 1\end{bmatrix} \begin{bmatrix}x_i \\y_i \\1\end{bmatrix}\tag{7}$$

- Rotación

Normalmente realizada sobre el origen de coordenadas, aunque a continuación se muestra la notación general para aplicarla sobre un punto cualquiera definido (8), un ángulo θ .

$$\begin{bmatrix}x_f \\y_f \\1\end{bmatrix} = \begin{bmatrix}1 & 0 & x_0 \\0 & 1 & y_0 \\0 & 0 & 1\end{bmatrix} \begin{bmatrix}\cos\theta & -\text{sen}\theta & 0 \\ \text{sen}\theta & \cos\theta & 0 \\ 0 & 0 & 1\end{bmatrix} \begin{bmatrix}1 & 0 & -x_0 \\ 0 & 1 & -y_0 \\ 0 & 0 & 1\end{bmatrix} \begin{bmatrix}x_i \\y_i \\1\end{bmatrix}\tag{8}$$

- Escalado

Aumenta o disminuye en la dirección horizontal y vertical según el valor de a y b sean menores o mayores que la unidad, respectivamente (9)

$$\begin{bmatrix}x_f \\y_f \\1\end{bmatrix} = \begin{bmatrix}1/a & 0 & 0 \\ 0 & 1/b & 0 \\ 0 & 0 & 1\end{bmatrix} \begin{bmatrix}x_i \\y_i \\1\end{bmatrix}\tag{9}$$

Las transformaciones presentadas son las más comunes, pero no las únicas.

En cualquier caso, es importante aplicar correctamente los algoritmos de interpolación para evitar valores erróneos en la imagen resultado, ya que las nuevas posiciones de los píxeles transformados en muchos casos pueden no ser valores enteros. Este hecho puede ocasionar que varios píxeles tomen el mismo píxel destino en la nueva imagen, o que algún punto no tenga valor asignado.

Existen algoritmos de interpolación propuestos por diversos autores [5]. Se van a presentar a continuación algunos de los más importantes:

- Vecino más próximo

Se trata de asignar el valor del píxel destino basándose en el más cercano de la imagen original al píxel. Este algoritmo tiene la ventaja de una velocidad alta. Sin embargo, la calidad final no es la más adecuada para ciertos objetivos, al ser menos preciso que otros.

- Interpolación bilineal

En este caso el nivel de gris se obtendrá de la media ponderada de los píxeles de la vecindad, en un entorno o matriz de 2×2 . La ponderación mencionada se traduce en un mayor peso de los píxeles según la cercanía espacial al original. El resultado es más preciso que empleando el algoritmo anterior (figura 21).

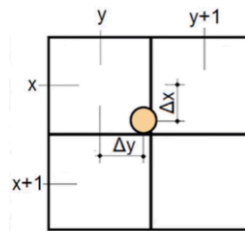


Figura 21. Interpolación bilineal [5]

- Interpolación bicúbica

Este sistema es similar al de la interpolación bilineal, pero en este caso la media ponderada se obtendrá de una matriz 4×4 , lo que suponen 16 píxeles, cuyo peso igualmente será mayor según la cercanía al punto estudiado (figura 22).

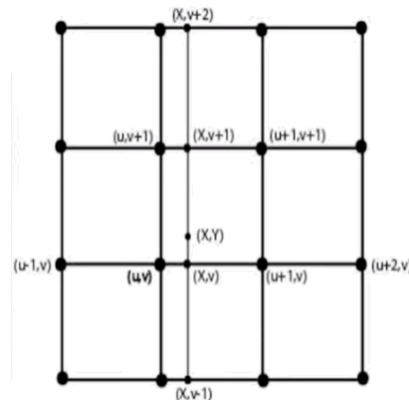


Figura 22. Interpolación bicúbica [5]

Correlación

La correlación entre dos funciones $f(x,y)$ y $h(x,y)$ es la que se presenta en (10).

$$g(x,y) = h(x,y) \circ f(x,y) = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} f^*(i,j)h(x+i,y+j) \quad (10)$$

El valor de la función correlación $g(x,y)$ es la multiplicación de ambas al superponerse. De esta forma, cuanto más coincidan ambas funciones, cuando se repita una parte de la función $h(x,y)$ en la función $f(x,y)$, más elevado será el resultado, siendo máximo cuando la coincidencia sea completa, y nulo en el caso de que esto no se cumpla en ningún punto. Es un método de medir la similitud.

Con la correlación se puede encontrar si una función está comprendida en otra que, traducido al caso de las imágenes, facilita encontrar en la imagen objetivo una imagen o patrón definido externamente.

Este método puede tener algunos inconvenientes. Por ejemplo, si un objeto engloba a otro tendrá la respuesta de que coincide perfectamente sin ser cierto. También posibles cambios de niveles de gris pueden ser un problema en la correcta obtención del resultado final. Por ello se aplica, la correlación normalizada, de acuerdo a la ecuación que sigue (11).

$$g(x,y) = \frac{\sum_i \sum_j (f(x+i,y+i) - \bar{f})(h(i,j) - \bar{h})}{\sqrt{\sum_j \sum_j (f(x+i,y+i) - \bar{f})^2 \sum_j \sum_j (h(i,j) - \bar{h})^2}} \quad (11)$$

Con esta herramienta pueden buscarse plantillas o patrones en el modelo. Cabe destacar, sin embargo, que este sistema no es inmune a rotaciones o escalas diferentes, es decir, si el objeto buscado es de tamaño diferente puede no encontrarlo correctamente, igual que si su posición geométrica se encuentra rotada y/o deformada. Únicamente no afecta la traslación simple.

En el lado izquierdo de la figura 23 se observa un cuadrado blanco sobre un fondo negro, y al aplicar la correlación con una figura idéntica, se obtendrá el resultado de la derecha en la misma figura, donde una mayor intensidad del blanco indica una mayor coincidencia o similitud, indicando en el punto central la completa coincidencia (cuando ambos cuadrados están superpuestos, el resultado es un valor de 255 en la escala de grises).



Figura 23. Correlación [5]

Reducción de ruido

Existe un defecto habitual en la toma de imágenes que aparece como píxeles con un valor de gris diferente del esperado, debido a una posible distorsión en el sensor de la cámara o a una alteración en la transmisión de la señal. Dichos defectos se conocen como ruido de la imagen, y puede ser de diferentes tipos, siendo el más común el conocido como ruido gaussiano, más uniforme; también tienen alta importancia el impulsional, cuyo resultado son píxeles con valores muy alejados de la realidad de sus vecinos, o el frecuencial debido a la interferencia de otra señal, y el multiplicativo por la multiplicación de dos señales diferentes, que se encuentran en la fila inferior de la figura 24 por orden de mención.

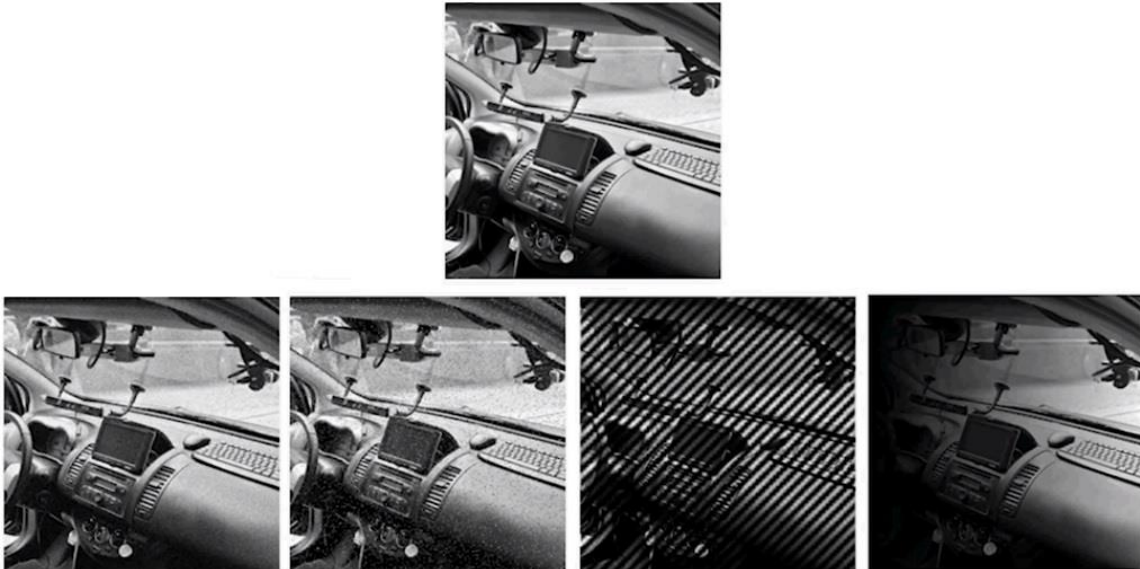


Figura 24. Tipos habituales de ruido [5]

A fin de eliminar dichos ruidos se emplean filtros paso bajo, de forma similar a como se explicó en la convolución, pero en este caso los filtros tomarán otros valores. El más sencillo es aplicar una matriz de 3x3 unitaria, que dé la misma importancia a todos los píxeles de la subregión. Si se quiere dar más importancia a los píxeles cuanto más cerca del centro, la forma general de dicha matriz será (con b un número natural definido):

1	b	1
b	b^2	b
1	b	1

Siendo la ganancia de cualquiera de estas matrices la unidad, para evitar variaciones en la imagen, en el caso unitario sería multiplicada por un noveno.

Otro filtro lineal habitual es el gaussiano, donde los elementos de la matriz vienen dados por la expresión siguiente (12), que da importancia exponencial a la cercanía al punto x,y en función de la varianza σ :

$$G(x,y) = e^{-\frac{(x+y)^2}{2\sigma^2}} \quad (12)$$

En cualquier caso, estos filtros deben ser aplicados con cuidado ya que pueden desdibujar bordes y eliminar objetos pequeños al homogeneizar regiones.

Realce y detección de bordes

En contraposición a los filtros para eliminar el ruido que se mostraban en el punto anterior, aquí se van a resaltar aquellos píxeles con valores diferentes de sus vecinos, es decir, se aplicarán ahora filtros que disminuyan las frecuencias bajas. El proceso es similar al de reducción de ruido pero, en este caso, la matriz aplicada tendrá un número positivo en el centro de la misma, y los demás valores serán negativos, ya sean de valor constante o aumentado progresivamente según cercanía espacial al número positivo central.

El trabajo de análisis más notable con respecto de los bordes es, en cualquier caso, la detección de estos, independientemente de si se ha facilitado la labor con el filtro pasa alto expuesto en el párrafo previo, los métodos de análisis son los mismos, que se explican a continuación.

Los bordes son identificados como los puntos donde la variación de intensidad es mayor, hablando en términos de nivel de gris. Esta identificación es realizada mediante el análisis de la derivada primera o de la derivada segunda.

Si se analiza la primera derivada, el gradiente, habrá que encontrar los máximos, que indicarán un cambio en la imagen. De esta manera se podrá obtener una solución con una cierta velocidad o eficiencia. Sin embargo, la precisión puede no ser la buena si la transición entre objeto y fondo no está bien definida.

Como simplificación del vector gradiente, dado el carácter discreto de las imágenes, se tiene la siguiente ecuación (13):

$$\nabla f(x, y) = \begin{bmatrix} G_x & G_y \end{bmatrix} = \begin{bmatrix} \frac{\Delta f}{\Delta x} & \frac{\Delta f}{\Delta y} \end{bmatrix} \tag{13}$$

Representado por las máscaras:

$$G_x = \frac{\Delta f}{\Delta x} \begin{bmatrix} -1 & 1 \end{bmatrix} * f(x,y) \qquad G_y = \frac{\Delta f}{\Delta y} \begin{bmatrix} -1 \\ 1 \end{bmatrix} * f(x,y)$$

Donde x representa la dirección horizontal del punto e y la vertical, siendo f la función gradiente.

Sin embargo, esta expresión primigenia fue pronto sustituida debido a su alta sensibilidad al ruido, por tener únicamente en cuenta un binomio de píxeles, por otras máscaras que al mismo tiempo que calculan el gradiente suavizan el resultado, a fin de ser menos sensibles a dicho ruido.

Las más habituales son la de Roberts (2x2), la de Prewitt (3x3) similar a la de Roberts, pero en un dominio mayor, la de Sobel (3x3) que da más importancia a los píxeles centrales que los diagonales, o la isotrópica (3x3) que detecta los bordes diagonales con mayor éxito. Una muestra matricial de todos ellos puede encontrarse [5].

Si se atiende a las derivadas de segundo orden o laplacianas, se emplean habitualmente las dos matrices que siguen (14), aplicadas sobre el píxel como se ha explicado en el apartado de convolución:

$$\nabla^2 f(x, y) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

-1	-1	-1
-1	8	-1
-1	-1	-1

0	-1	0
-1	4	-1
0	-1	0

(14)

Si bien existen variaciones de la laplaciana (ya que el método en sí mismo es poco empleado debido a la sensibilidad al ruido), se va a mencionar a continuación un método de los más utilizados, el detector de Canny, que se basa en tres condiciones [5]:

- Error: Se deben detectar todos y sólo los bordes
- Localización: La distancia entre el pixel señalado como borde y el borde real debe ser tan pequeña como se pueda
- Respuesta: No debe identificar varios pixeles como bordes cuando sólo exista uno

Y que expresa analíticamente como se muestra a continuación (15), cuyo desarrollo y explicación de los factores implicados puede obtenerse en [23].

$$SNR = \frac{A \left| \int_{-W}^0 f(x) dx \right|}{n_0 \sqrt{\int_{-W}^W f^2(x) dx}} \quad \text{Localización} = \frac{A |f(0)|}{n_0 \sqrt{\int_{-W}^W f^2(x) dx}} \quad \text{Distancia} = \pi \left(\frac{\int_{-\infty}^{\infty} f^2(x) dx}{\int_{-\infty}^{\infty} f'^2(x) dx} \right)^{\frac{1}{2}} \quad (15)$$

El operador óptimo propuesto para el método por el propio Canny es la derivada de una gaussiana.

En la figura 25 puede verse la efectividad del sistema, donde se detecta el cambio entre fondo y grano de arroz (lado izquierdo de la figura) marcando así esos puntos, que formarán los bordes mostrados en el lado derecho de la figura.

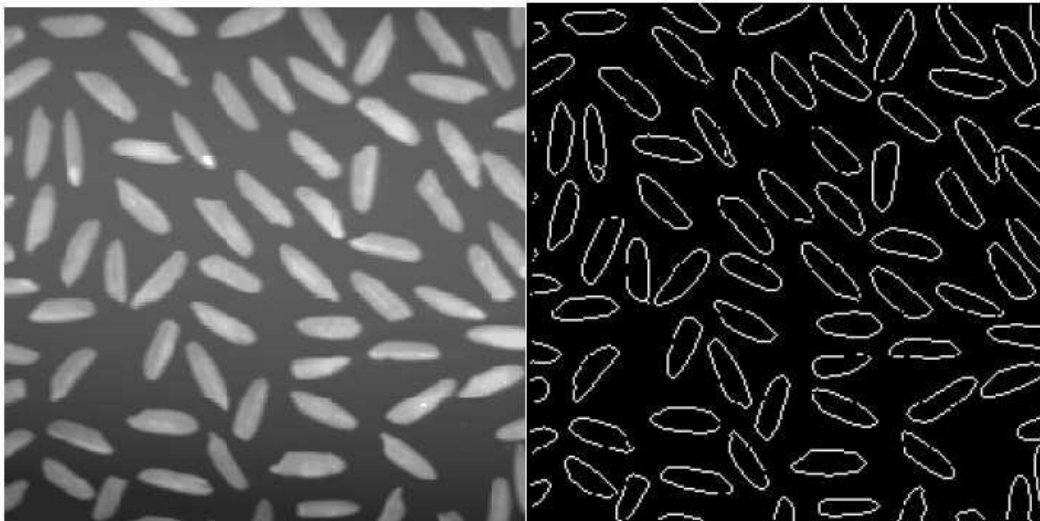


Figura 25. Método de Canny [10]

Otros métodos de análisis de la imagen

Se han presentado las herramientas y métodos de análisis de imágenes cuya utilidad directa es bastante relevante, y que son además de uso frecuente.

Hay, sin embargo, un gran abanico de posibilidades no presentadas en este resumen pero que en caso de ser útiles para el proyecto se estudiarán con más atención en el apartado correspondiente. Por mencionar algunos, es común trabajar con las texturas de los objetos mediante gradientes, normalmente orientados a fin de identificar las mismas.

Otra herramienta es la llamada matriz de concurrencia de niveles de gris (MCNG), que indica el número de veces que aparecen en la imagen píxeles de un valor determinado cuyo píxel vecino es de otro valor también definido, en una determinada dirección. Esta matriz ayuda a calcular valores típicos como distribuciones, contraste, correlación y otros menos comunes como la entropía. Estos valores son usados principalmente para la clasificación de imágenes e incluso de texturas.

Por último, relacionado tanto con estos conceptos como con lo anteriormente expuesto, se emplean para identificar objetos o regiones en la imagen los llamados clasificadores, que son vectores de características empleados para definir una imagen. Dichos vectores pueden incorporar valores como la distribución, el contraste, etc. y a partir de ellos comparar las características con otros vectores de la propia imagen, o incluso de una base de datos para clasificar el resultado.

Página en blanco intencionadamente

3. CLASIFICACIÓN DE LANGOSTINOS

3.1. MATERIALES: ESPECIES DE LANGOSTINO

Existen una gran variedad de especies de langostino en los océanos del mundo, y tomar en cuenta todos, además de costoso, sería ineficiente, pues aumentaría la incertidumbre y la probabilidad de fallo en aguas donde posiblemente ciertos tipos de langostino no se encuentren y otros sí.

Dada la naturaleza de este proyecto, donde se pretende demostrar la eficacia de un sistema de visión artificial para clasificar las especies, se va a tener en cuenta un grupo de ellas que se encuentren en aguas cercanas, en concreto, en el litoral este del continente africano.

Las especies se mencionan con una breve descripción y una foto para ilustrarlas.

Para la realización de este proyecto se dispone de 50 fotografías por tipo de langostino, tomadas con una cámara Canon 4000D como se mencionó en el capítulo de introducción, tomadas a una distancia focal de 750 mm. Las imágenes se encuentran en formato JPEG a un tamaño reducido de 600x400 píxeles.

- **PK: Penaeus Kerathurus**, también conocido como langostino tigre. Su tamaño varía entre los 19 y los 22 cm de media, y vive a una profundidad menor de 100 metros. Se caracterizan por tener zonas rayadas de color anaranjado en su cuerpo, normalmente en formas redondeadas (figura 26).



Figura 26. *Penaeus Kerathurus*

- **PM: Penaeus Monodon**, el que se presenta como esta especie en el proyecto no es exactamente el auténtico *Penaeus Monodon*, que se conoce como Jumbo. Sin embargo, es común encontrar a esta especie, pariente del Jumbo. Algo más pequeños que los PK; muestran un cuerpo de color más homogéneo y pardo (figura 27).



Figura 27. *Penaeus Monodon*

- **PP: Penaeus Peche**, también conocido como Indicus, con un tamaño medio de unos 17 cm, presenta un color homogéneo anaranjado algo más intenso que las otras especies (figura 28).



Figura 28. Penaeus Peche

- **PV: Penaeus Vannamei**, que, si bien pertenece a aguas pacíficas, se ha tomado en cuenta por existir una variedad muy similar en aguas índicas, y ser más sencillo de conseguir. Su tamaño medio es algo más reducido que las otras especies presentadas, aunque se han dado casos de hasta 23 cm. Vive en profundidades de hasta 72 metros, y su color es blancuzco, y en general homogéneo, si bien puede presentar un ligero moteado oscuro en el cuerpo (figura 29).



Figura 29. Penaeus Vannamei

- **J: Jumbo**, es la especie Penaeus Monodon, pero en este proyecto se ha considerado por su nombre común. Esta es sin duda la especie más grande de langostino, con tamaños de hasta 33 cm. Su color es notablemente más oscuro, y presenta un rayado intenso, normalmente compuesto por una franja oscura y una clara seguidas (figura 30).



Figura 30. Jumbo

3.2. IDENTIFICACIÓN DEL LANGOSTINO

El proceso de identificación del langostino se ha dividido en tres fases: una primera fase de discretización o umbralización que separa el espécimen del fondo, una segunda fase en la que se extraen las características deseadas del langostino, y una fase final donde se definen los vectores de características y se clasifica la imagen en función de estos, obteniendo resultados (figura 31).

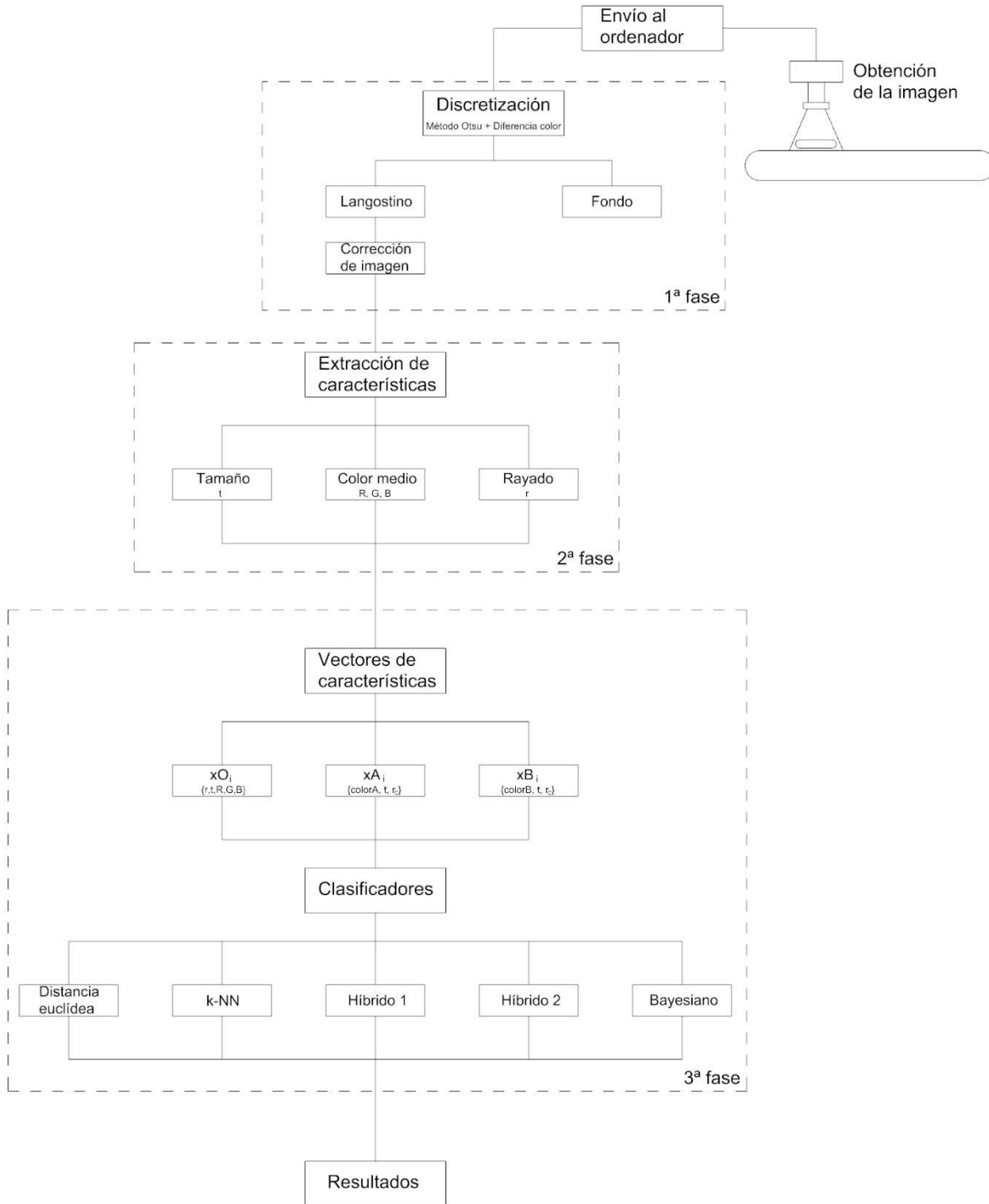


Figura 31. Esquema del proceso

3.2.1. PRIMERA FASE: SEGMENTACIÓN

Las fotografías/videos tomados por los equipos de adquisición de imagen idealmente mostrarán un espécimen centrado, con la orientación esperada, sin defectos y solitario sobre el fondo blanco en el que se encuentra. Sin embargo, esto no va a ser así en la mayoría de los casos, donde podrán encontrarse algunos defectos e incluso interferencias en la imagen.

Antes de entrar en el tema de posición u orientación y su influencia en el análisis, conviene identificar dónde se encuentra el langostino en la imagen y tratar de aislarlo de todo aquello que no sea de interés para la clasificación. Con este punto se hace referencia a posible ruido de la imagen, así como defectos en el fondo sobre el que se encuentra el espécimen e incluso restos de otros especímenes o elementos ajenos.

Por ello, deberán realizarse una serie de trabajos de procesado de la imagen para poder aislar e identificar el sujeto de cualquier otra cosa que no sea de interés para el estudio de éste.

El esquema de trabajo será el presentado en la figura 31, y que en la figura 32 se amplía sobre la zona que va a trabajarse ahora: la primera fase.

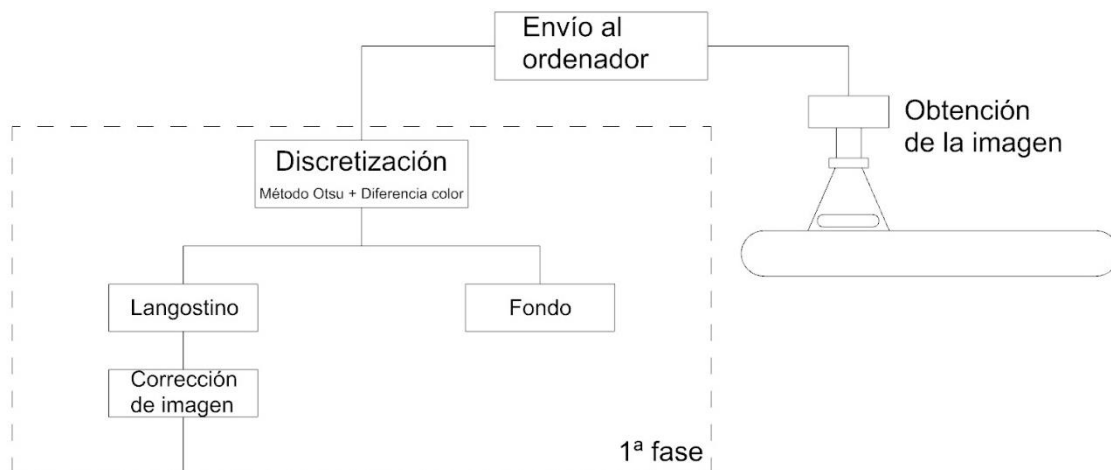


Figura 32. Esquema trabajo primera fase

Como se ha explicado en el capítulo 2, cualquier imagen, se convierte en información para ser tratada de forma matricial, y dado que vamos a trabajar en el espacio de color RGB, será así como la obtengamos, como un vector de tres matrices, cada una correspondiente a uno de estos colores. No obstante para una primera definición de la región que ocupa el sujeto a analizar se empleará una imagen convertida a escala de grises, que supone una única matriz cuyos píxeles toman valores entre 0 y 255 (figura 33).

Este primer paso de procesado se puede realizar de forma sencilla a través del comando perteneciente al Image Processing Toolbox de Matlab.

```
ImagenGris = rgb2gray(ImagenRGB);
```

O en su defecto, el cálculo directo que realiza este comando, de acuerdo con la normativa NTSC (16).

$$\text{ImagenGris} = 0.2989 * R(:, :) + 0.5870 * G(:, :) + 0.1140 * B(:, :) \quad (16)$$

Se tendría entonces una matriz, con la que poder trabajar de forma más eficiente en la primera fase de discretización.

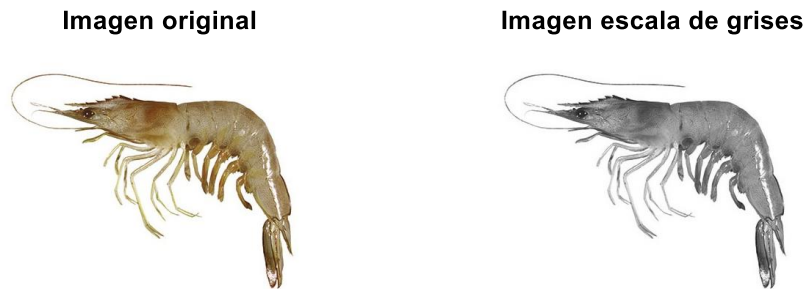


Figura 33. Imagen escala de grises

Separación del langostino y fondo

El siguiente paso a realizar será la separación del fondo respecto del propio objeto que se encuentra sobre él, y que dependerá del tipo de dicho fondo. En el caso más común son cintas de transporte, de color blanco o marfil, cuyos valores en la escala de grises corresponden a valores entre 220 y 255, suponiendo un estado de limpieza ideal.

Conociendo estos hechos, parece lógico tomar valores en torno a 200 para discernir el fondo del objeto, de tal forma que para cada píxel de la imagen se le aplique dicho umbral. Si está por encima significará que el píxel pertenece al subconjunto fondo de la imagen, caso contrario su pertenencia será al subconjunto no fondo.

Para realizar este simple cálculo se escribe una función en Matlab de tal forma que sus entradas sean la imagen en escala de grises y el valor umbral que hayamos determinado.

```
function Id=umbral(I,U)
dimImg=size(I);
Id=zeros(dimImg(1),dimImg(2));
for i=1:dimImg(1)
    for j=1:dimImg(2)
        if I(i,j)>U
            Id(i,j)=0;
        else
            Id(i,j)=255;
        end
    end
end
```

De esta manera la matriz *Id* resultado de dicha función estará compuesta de dos valores, 0 para todos los puntos no pertenecientes al objeto, y 255 para aquellos que sí. Es una matriz que puede normalizarse en ceros y unos sin perder información.

Continuando con la imagen de la figura 33, se muestran en la figura 34 el resultado de aplicar esta función para diferentes valores de umbral, U.

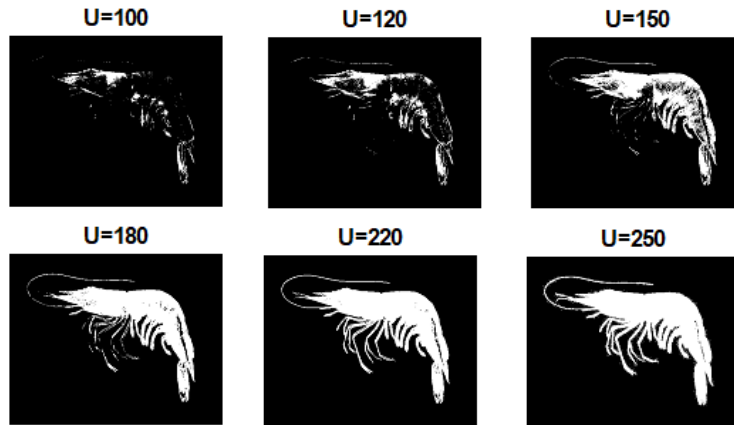


Figura 34. Umbralización con diferentes valores fondo blanco

Como conclusión inmediata y trivial, con un valor de umbral muy reducido se perderá información sobre el objeto al ser considerado fondo, mientras que, para valores altos del umbral, apenas hay información falsa. Solo para un umbral de casi el valor máximo (250 de 255), en la zona de las patas traseras puede observarse que han sido tomadas como langostino zonas que en realidad no pertenecen a dicho subconjunto. Esta observación se realiza con una imagen en la que el fondo es idealmente blanco, y la iluminación correcta, ya que tan solo una pequeña región de la zona de la cola tiene una intensidad inusualmente elevada debido al reflejo de la luz.

Tomamos ahora un ejemplo de otro espécimen, esta vez sobre un fondo que no es idealmente blanco, corresponde a un fondo más realista, con un tono marfil.

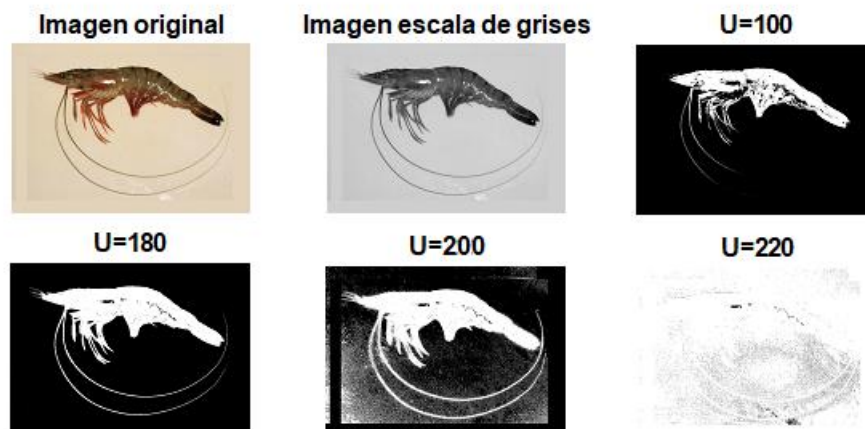


Figura 35. Umbralización con diferentes valores fondo coloreado

Ahora el resultado es algo diferente (figura 35)., lo primero a destacar es la menor pérdida de información para valores bajos frente al ejemplo anterior, debido a una imagen más oscura, tanto por iluminación como por el propio color del animal. El punto clave sin embargo se encuentra en el fondo sobre el que descansa la especie, cuyo valor medio se encuentra más cercano al 200 que, como en el caso anterior, al 255.

Además, el fondo no es perfectamente liso, lo que significa que esa rugosidad añadida debido a la textura del propio elemento genera algunas diferencias de iluminación en él, algo cuya traducción analítica es una serie de valores no uniformes en su extensión. Por eso, a partir del valor $U=200$, una gran cantidad de píxeles de la imagen dejaría de pertenecer al subconjunto fondo de la imagen y pasarían en cambio a pertenecer al objeto.

Para valores más elevados, como por ejemplo $U=220$, resulta evidente que la información resultante no es útil desde el punto de vista de identificación del objeto langostino.

Se deduce de todo lo anterior la importancia de definir un buen valor umbral en la identificación de la especie, que, de acuerdo con todas las diferentes imágenes de estudio disponibles, de forma empírica es $U=180$, en gran parte de los casos. Sin embargo, hay un caso que se ha estudiado y que no es único en el que el valor umbral necesario para una correcta resolución se alejaba del resultado deseado, pues los valores de la matriz eran en general bastante elevados.

Debe buscarse por lo tanto una solución para el cálculo de umbral de tal forma que pueda automatizarse la obtención de éste. Existen diversos métodos propuestos por autores para esta finalidad, y entre ellos el seleccionado en el presente proyecto es el conocido como método de Otsu [10], que se basa en maximizar la varianza entre clases, de acuerdo con la frecuencia de repetición de cada nivel de gris, dividido por el número total de píxeles en la imagen. Esto dará la probabilidad de ocurrencia de un determinado nivel de gris.

Si se desea profundizar en el desarrollo del método se encuentra en [10]. Para su aplicación al presente proyecto se empleará el comando integrado en la librería para procesamiento de imágenes de Matlab: *graythresh(I)*, donde *I* es una imagen en escala de grises. El resultado será un valor normalizado entre 0 y 1; habrá que multiplicarlo por 255 para usarlo en la función ya descrita.

En la figura 36 se muestra la imagen previamente mencionada que ofrecía un resultado poco práctico con $U=180$, y la opción presentada mediante el método de Otsu propuesto.

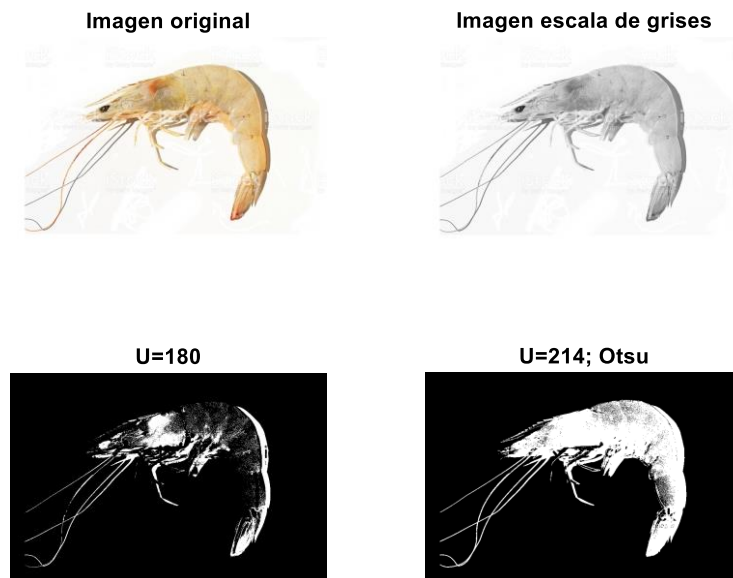


Figura 36. Umbralización con Otsu

Se ha presentado una solución eficiente y ajustable para discretizar el objeto del fondo, pero esta solución simple puede resultar algo imprecisa si en vez de ser el fondo habitual de cintas claras el animal se encuentra sobre una superficie metálica de acero inoxidable, lo que, si bien es menos frecuente, también es una realidad extendida.

Por ejemplo, la figura 37 muestra un langostino sobre bandeja metálica iluminado por un foco desde el margen izquierdo de la imagen.



Figura 37. Umbralización simple fondo metálico

Tras realizar la conversión a escala de grises, por ser el fondo una superficie de valor no homogéneo con predominancia de tonos azulados, si la iluminación es escasa, el resultado será un fondo de diferentes valores de gris que se confundirán con la escala de grises del langostino. Así pues, al ejecutar la función *umbral* definida previamente, aún con el valor U determinado mediante el método de Otsu, tendríamos que en dicha imagen (figura 37), gran parte del fondo pertenece al mismo subconjunto que el propio langostino, mientras que tan solo la parte claramente iluminada por el foco pertenece al subconjunto fondo que le corresponde.

Lo que se deduce de la imagen es que, debido a la característica reflectante del metal, mucho mayor que la del propio espécimen, con una iluminación adecuada este problema quedaría probablemente resuelto. En cualquier caso, una posible mejora de la función discretizadora sería tener en cuenta también las características de color de la imagen, es decir, trabajar con el vector de matrices RGB.

Se puede observar que las dos matrices de colores de mayor relevancia serán la de tonos rojos R , cuyos valores más elevados pertenecerán al langostino en la mayoría de los casos, y la de tonos azules B , donde se encuentra el color metálico de fondo.

Se propone entonces para separar fondo de objeto, en vez de un sistema basado en un umbral fijo de nivel de gris, realizar dicha separación de acuerdo con la predominancia de tonalidades rojas o azules en el vector RGB de cada píxel. Para este objetivo se realiza una simple diferencia de valores: $R-B$; de esta manera, un resultado negativo indicaría predominancia del tono azul y por lo tanto pertenencia al subconjunto fondo, mientras que un resultado positivo supondría su pertenencia al subconjunto langostino, sin necesidad de definir valores umbral.



Figura 38. Umbralización por color fondo metálico

La nueva función, que se ha llamado `umbralC` realizará este sencillo cálculo para cada píxel, empleando ambas matrices y decidiendo a que conjunto pertenece en función del signo del resultado obtenido en la resta (figura 38).

El resultado de este procesado dista notablemente del obtenido mediante el sistema previamente diseñado basado en el umbral de la escala de grises, ofreciendo una imagen mejor umbralizada. Sin embargo, hay que destacar dos posibles problemas de este código; uno es para un caso como el ejemplo anterior (figura 35), donde el fondo toma tintes rojizos, ya que entonces la predominancia será claramente de tonalidades rojas frente a tonalidades azules, y el resultado de dicha función será erróneo ya que todo píxel de la imagen se habría considerado como langostino, fondo incluido, y sería por tanto una imagen enteramente blanca (figura 39).



Figura 39. Umbralización por color fondo coloreado

Otro posible problema podría venir dado por un exceso de iluminación en la imagen, ya que en los puntos de iluminación máxima $R=G=B=255$, con lo que la diferencia sería nula y por tanto pertenecería al fondo. Hay puntos donde la elevada intensidad podría confundir los tonos y dar resultados erróneos o poco precisos (figura 40).



Figura 40. Umbralización por color fondo metálico 2

Función `umbralC`:

```
function Id=umbralC(original,U)
dimImg=size(original);
Id=zeros(dimImg(1),dimImg(2));
Idouble=im2double(original);
for i=1:dimImg(1)
    for j=1:dimImg(2)
        if (Idouble(i,j,1)-Idouble(i,j,3))<=0 % R-B
            Id(i,j)=0;
        else
            Id(i,j)=255;
        end
    end
end
```

Se trata de encontrar una posible solución a ambos problemas. Por un lado, a las imágenes donde el fondo sea de material plástico, con tonos blancos o marfil, y por otro, a aquellas imágenes en las que el fondo empleado sea metálico, y la iluminación no sea la adecuada.

La mejora añadida sobre el código anterior se encuentra remarcada en negrita. Se añade una línea extra donde se evalúa el umbral para la escala de grises; de tal manera que, si la diferencia de tonalidades no indica una mayor presencia de azul, se realizará una segunda evaluación del píxel de acuerdo con la condición de umbral sobre la imagen en escala de grises. El único error podría venir dado por un langostino con mayor presencia de azul que de rojo en su tonalidad, lo cual, debido a las especies de langostinos estudiadas en el proyecto (así como el resto de las especies que conviven con éstas en la naturaleza), no es motivo de preocupación por su probabilidad de ocurrencia despreciable.

Nótese también que ahora la función tiene una nueva entrada, consistente en la imagen original con sus tres matrices de color: R, G, B.

```
function Id=umbralCv2(I,U,original)
dimImg=size(original);
Id=zeros(dimImg(1),dimImg(2));
Idouble=im2double(original);
for i=1:dimImg(1)
    for j=1:dimImg(2)
        if (Idouble(i,j,1)-Idouble(i,j,3))<=0 % R-B
            Id(i,j)=0;
        elseif I(i,j)<U
            Id(i,j)=255;
        end
    end
end
```

Aplicando esta nueva función a los cinco ejemplos mostrados hasta ahora podemos ver (figura 41), que allí donde la función umbral sobre escala de grises falle, la alternativa basada en la diferencia de las matrices roja y azul ofrece mejor resultado, y viceversa. Tan solo hay que decidir cuál aplicar en cada caso.

También se desprende de los resultados que la combinación de ambas ofrece buenos resultados en todos los casos. Luego a priori parece que la función discretizadora a emplear de forma general será esta última, ya que es más inmune a una iluminación poco adecuada y a un cambio de fondo del langostino entre los habituales.

En los tres primeros casos la solución final es prácticamente igual que el cálculo realizado con la función umbral sobre la escala de grises. En el tercer caso es, por el contrario, similar al que ha empleado la función diferencia de matrices de colores azul y roja. El último caso, sin embargo, presenta una solución conjunta mucho más acertada que cualquiera de sus dos versiones previas (umbral de grises o diferencia de rojo y azul).

Por lo tanto, en base a estos cuatro ensayos podría decirse que, si a priori desconocemos el tipo de imagen con la que estamos trabajando, su iluminación, o la base en que se encuentra apoyado el langostino:

- La función umbral en escala de grises ha ofrecido un 60% de resultados buenos y 40% de resultados inaceptables

- La función diferencia de matrices de color roja y azul resulta en un 20% de resultados buenos, un 40% de resultados aceptables y un 40% de resultados inaceptables
- La función conjunta ($Umbral \cap R - B$) ofrece resultados buenos en el 100% de las muestras estudiadas

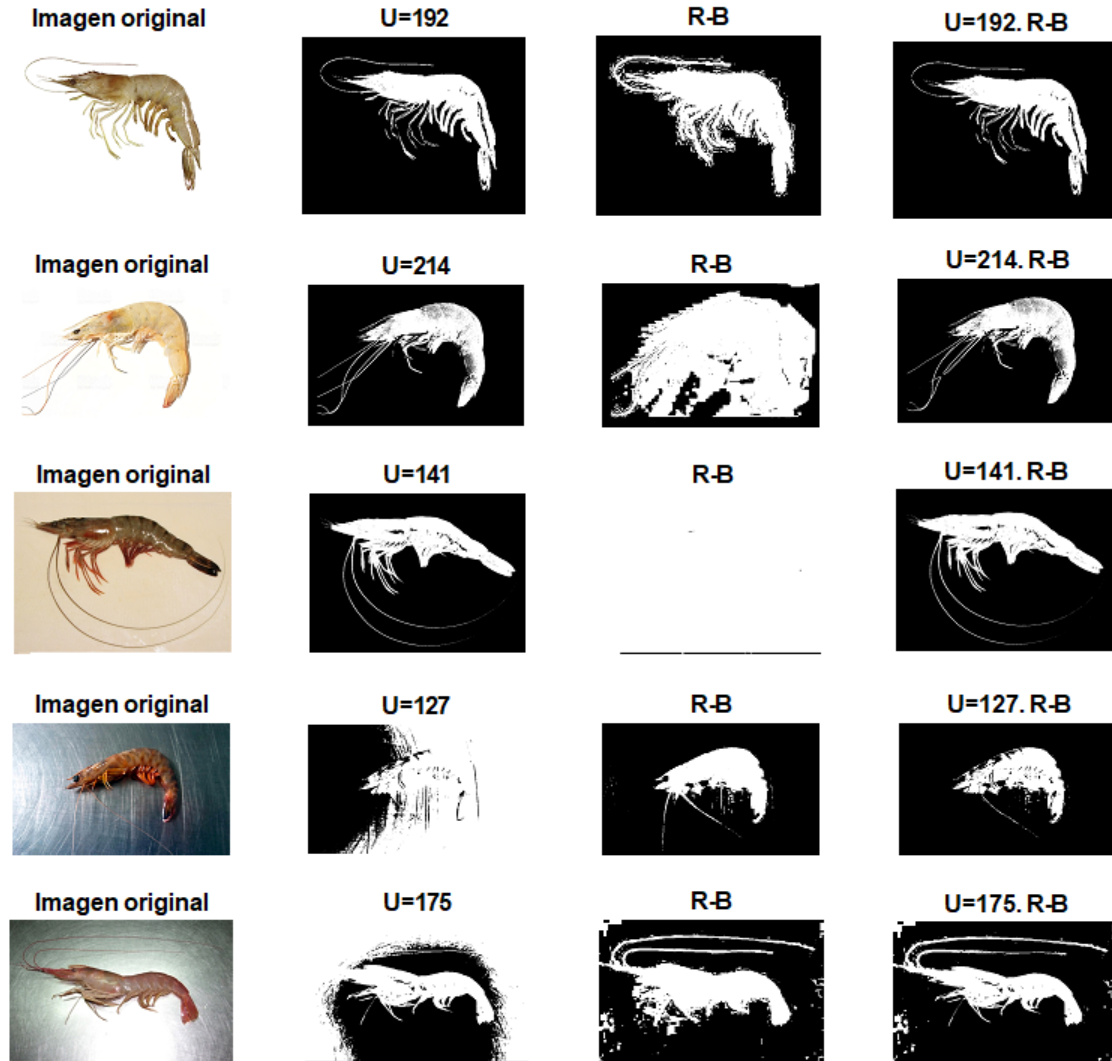


Figura 41. Comparativa umbralizaciones

Reconstrucción y filtrado del espécimen

Hasta ahora se ha mostrado el procedimiento desarrollado para diferenciar el langostino del fondo.

Como se puede ver en las figuras anteriores, aunque la imagen resultante en la discretización permite apreciar a simple vista la plantilla o forma del langostino, existen varias regiones en su interior que son erróneamente clasificadas como fondo, debido sobre todo al proceso de umbralización en escala de grises, ya que zonas con alta intensidad lumínica en el espécimen son tomadas como fondo, dando lugar a estas anomalías.

Por otro lado, el error inverso también puede estar presente, es decir, que se clasifique como langostino parte del fondo cuando no es así.

Para evitar ambos problemas se debe desarrollar un código que permita filtrar los píxeles de la imagen para asignarles su valor más probable (langostino o fondo), en función de sus píxeles vecinos. Dicho filtro se escribirá además con vistas a que elimine no únicamente estos píxeles aislados, si no también cualquier elemento menor que no forme parte del espécimen evaluado, como por ejemplo una pata o antena desprendida del langostino, o bien otro resto menor.

En el código que se presenta se define una función en la que las entradas son:

- I_d : la imagen binaria discretizada obtenida en el paso anterior, es decir aquella en la que los píxeles pertenecientes al subconjunto langostino valen 1 y los pertenecientes al subconjunto fondo valen 0.
- a : el valor de ajuste para definir el número total de vecinos a tener en cuenta, aplicado sobre la dimensión máxima de la imagen evaluada.
- p_F : el valor límite de creencia de pertenencia del píxel al subconjunto fondo, en función de la situación de pertenencia de sus píxeles vecinos. De tal forma que, si la creencia de que pertenece a dicho subconjunto es superior al límite definido, se considerará que efectivamente pertenece a éste.
- p_L : el valor límite de creencia de pertenencia a del píxel al subconjunto langostino, de igual manera que en el caso explicado para fondo (p_F).

Los tres valores de ajuste presentados pueden afinarse para cada muestra, buscando el valor óptimo. Sin embargo, se va a buscar empíricamente unos valores por defecto cuyos resultados sean lo más generales posibles. Evidentemente si se aplica el código sobre una instalación fija (tamaño de imagen, posición de cámara e iluminación considerablemente constante) sería conveniente ajustar estos valores en función de dichas características.

Debe destacarse también que los valores límite de creencia de pertenencia a un subconjunto pueden ser diferentes, de tal forma que una serie de píxeles se encuentren en una región de incertidumbre donde no se consideren que pertenecen a un subconjunto con una certeza mayor que la estipulada (p_F y p_L); por lo tanto, se quedarán con la situación de pertenencia original.

Todo píxel fuera de los márgenes de la propia imagen será considerado como fondo (no langostino).

Además de la imagen filtrada y corregida, la función también devuelve otra imagen, I_p , cuyos valores son la probabilidad de que un píxel sea o no langostino, normalizada entre 0 y 1.

```

function [Ir, Ip]=region(Id,ar,pF,pL)
dimImg=size(Id);
dA=round(ar*max(size(Id))); % Ancho y largo de región a analizar
totArea=(2*dA+1)*(2*dA+1); % Área total de región analizada
Irl=logical(Id); % Matriz de imagen filtrada
Ir=Irl;
Ip=zeros(dimImg(1),dimImg(2)); % Matriz de pertenencia
for i=1:(dimImg(1))
    for j=1:(dimImg(2))
        k=0;
        for il=(i-dA):(i+dA)
            for j1=(j-dA):(j+dA)
                if il<1 || j1<1 || il>dimImg(1) || j1>dimImg(2) ||
Irl(il,j1)==0
                    k=k+1; % Si NO es langostino
                end
            end
        end
        if (1-k/totArea)<pF % Es fondo
            Ir(i,j)=0;
        elseif (1-k/totArea)>pL % Es langostino
            Ir(i,j)=1;
        end
        Ip(i,j)=(1-k/totArea); % Construcción mapa de pertenencia
    end
end
end

```

Por ejemplo, un caso ya visto en el presente documento (figura 33) pero con un añadido en la imagen, la pata de otro espécimen (figura 42).

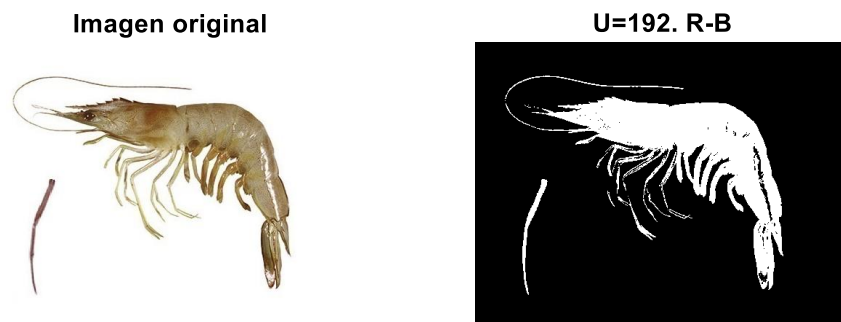


Figura 42. Langostino umbralizado con objetos indeseados

El objetivo es ahora encontrar la plantilla o silueta del animal, es decir, eliminar la pata no deseada de otro sujeto, rellenar las regiones consideradas como fondo dentro del propio langostino y, en la medida de lo posible, simplificar la imagen suprimiendo detalles como antenas o patas que a priori no van a emplearse en la metodología propuesta para la clasificación.

El resultado de aplicar la función descrita sería, en primer lugar, lo que se ha denominado mapa de pertenencia, debido a que sus valores entre 0 y 1 indican qué probabilidad de pertenencia al subconjunto langostino tiene cada punto de la imagen, siendo 1 la más alta y 0 la más baja.

Evidentemente este resultado variará en función del valor “*ar*”, anteriormente descrito, que define el número de vecinos a tener en cuenta para cada píxel, siendo su valor función porcentual del lado máximo de la imagen.

La figura 43 muestra dicho mapa de pertenencia para diferentes valores de este parámetro sobre la imagen previa (figura 42).

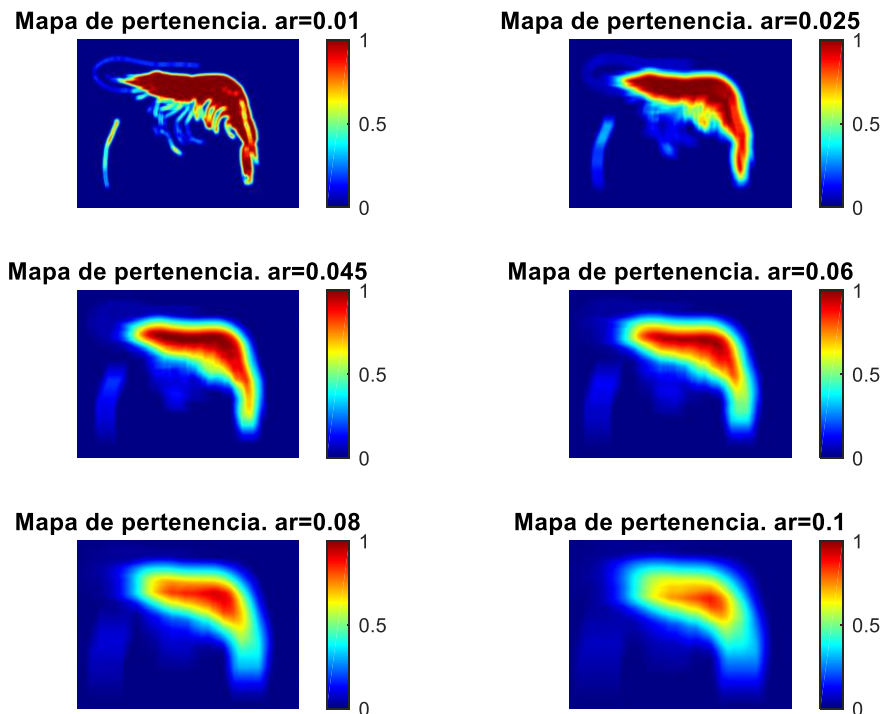


Figura 43. Mapas de pertenencia

Un valor de “*ar*” reducido tendrá en cuenta el valor de los vecinos más cercanos y apenas tendrá influencia sobre el resultado. Por ejemplo, en la primera imagen de la serie, donde el área de vecindad es un 1% de la dimensión máxima, se ve claramente que la pata ajena de la izquierda tiene una probabilidad elevada de pertenecer al langostino, mientras que la zona de la cola donde se reflejaba la luz tiene una probabilidad baja de pertenecerle; es decir, es un resultado que apenas aporta información adicional respecto de lo que ya se tenía en la instancia anterior.

Por el contrario, si el parámetro mencionado tiene un valor relativamente elevado, como por ejemplo mayor del 6% de la dimensión máxima en el área de vecindad estudiada, el resultado es difuso ya que, si bien las zonas con alta probabilidad de pertenecer al subconjunto langostino es casi seguro que pertenezcan, una gran área de éste ofrecerá un valor que indicaría una baja probabilidad de pertenencia, cuando no es así.

Valores intermedios, como 2,5% y 4,5%; serían más adecuados para el análisis, ya que aumentan las probabilidades de pertenencia de los píxeles del langostino antes no tenidos en cuenta, y al mismo tiempo disminuyen la probabilidad de pertenencia en aquello que, efectivamente, no pertenece. Todo ello sin que se observe una pérdida de información notable en la figura.

Se define, aunque sujeto a modificaciones en función de las condiciones ambientales de adquisición de las imágenes, un valor por defecto de $\alpha=0,03$.

De esta forma, siguiendo con el mismo ejemplo tendríamos el resultado de la figura 44.

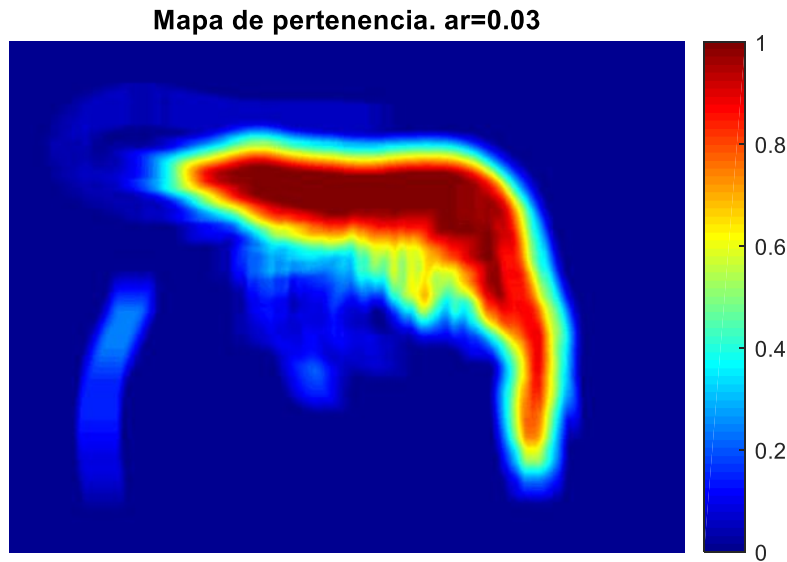


Figura 44. Mapa de pertenencia definido

En la función descrita existen otros dos parámetros configurables que usaremos para definir qué parte de la imagen es langostino y cuál fondo: los valores límite para cada píxel que permiten decidir si estos pertenecen a un subconjunto u otro en función de la probabilidad de pertenencia.

Dichos parámetros son μ_F y μ_L . El primero será el límite inferior a partir del cual cualquier píxel que tenga un valor normalizado inferior a dicho parámetro será considerado como fondo. El otro valor, μ_L , será el límite superior a partir del cual cualquier píxel que supere dicho límite con su valor normalizado será considerado como langostino. Por último, todo valor que se encuentre comprendido entre ambos límites no se verá modificado, manteniendo por tanto su valor de la matriz previamente discretizada "Id", al encontrarse dentro del intervalo de incertidumbre.

Al observar el mapa de pertenencia se puede deducir qué valores límite inferiores por encima de $\sim 0,3$ eliminarían píxeles aislados o pequeños objetos externos indeseados, que antes de aplicar este filtro no era diferente del resto del espécimen. Igualmente se limpia la propia imagen del langostino en cuanto a patas o antenas se refiere, lo que en principio no forma parte del presente proyecto.

La silueta del langostino tiene un valor entre 0,4 y 0,6; luego ajustando el límite inferior a uno de estos valores será identificada la figura del objeto, de forma más restrictiva cuanto mayor sea el valor, y viceversa cuando dicho límite decae.

Por último, atendiendo al límite superior, queda claro que por encima de $\sim 0,62$ (amarillo en la figura 43), los píxeles pertenecen al sujeto, pero debe tenerse en cuenta que un valor de pL muy elevado supondría que algunos puntos que sí pertenecen al langostino, como por ejemplo aquellos en la cola donde se reflejaba la luz en la imagen inicial, no sean tomados como tal, por ser una condición excesivamente restrictiva.

Algunos ejemplos sobre la imagen de estudio para diferentes valores de pF y pL , con ar fijo, se muestran en la figura 45.

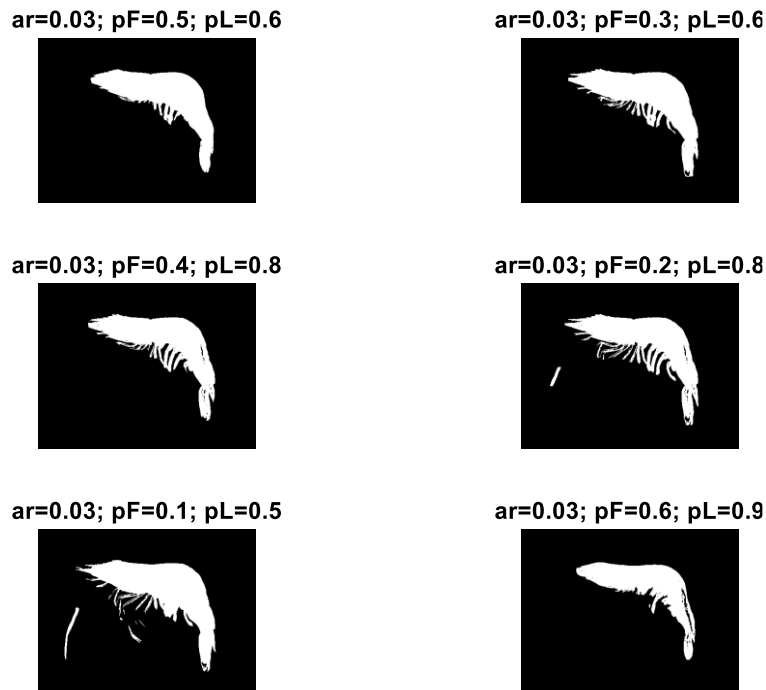


Figura 45. Imagen discretizada para diferentes límites

Se ha definido por defecto un valor de $pF= 0,3$ y $pL=0,8$; permitiendo un ajuste más adecuado cuando se conozca la situación externa o ambiental en que se encuentra.

Un resumen de los pasos realizados por el programa hasta ahora se encuentra en la figura 46.

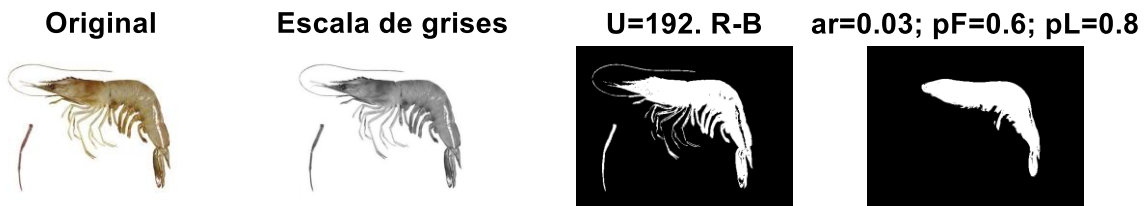


Figura 46. Pasos realizados por el programa en discretización [Elaboración propia]

Bordes

Si se hace una modificación en el código presentado de forma que, para ambas condiciones (píxel por encima y píxel por debajo de los límites establecidos) ofrezcan un resultado nulo, es decir, pertenencia al fondo, podríamos discretizar con esta misma función los límites o bordes de la figura, de mayor o menor grosor según los límites que definamos mediante los parámetros correspondientes.

Los resultados de esta pequeña modificación del código para diferentes valores pueden observarse en la figura 47.

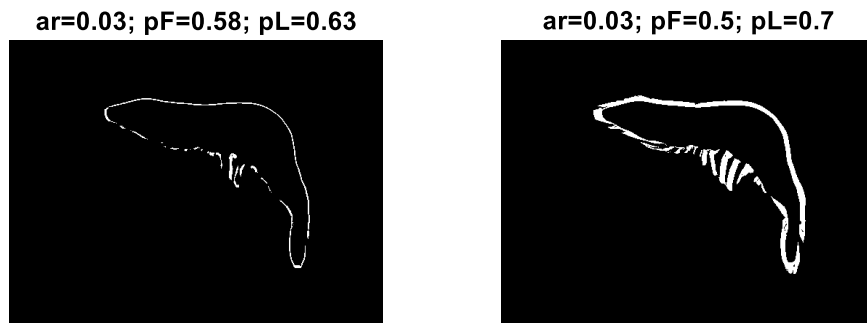


Figura 47. Extracción de bordes

3.2.2. SEGUNDA FASE: EXTRACCIÓN DE CARACTERÍSTICAS

En la primera fase se ha desarrollado un método de umbralización y discretización para los langostinos que ofrece a priori unos resultados suficientemente buenos para cualquier tipo de langostino sobre los fondos habituales.

En la figura 48 se tiene una ampliación del esquema general, correspondiente a la segunda fase.

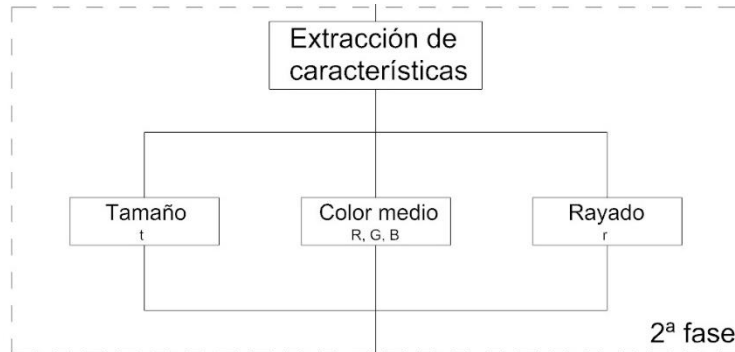


Figura 48. Esquema trabajo segunda fase

Si se aplica la plantilla consistente en la imagen binaria calculada en la primera fase, sobre la imagen original, se tendrá un correcto aislamiento del espécimen sobre el fondo y eliminación de los elementos no deseados. Siguiendo con uno de los ejemplos anteriores, se muestra el resultado en la figura 49.

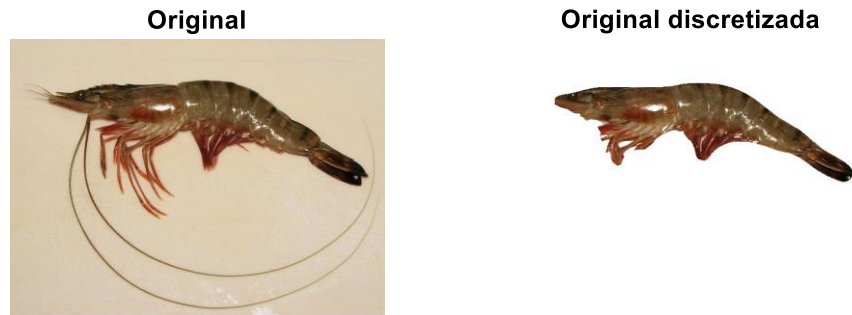


Figura 49. Imagen procesada

Ahora, sobre la imagen nueva vamos a extraer las características propias del animal, ordenándolas en un vector de características para, mediante un clasificador apropiado, identificar el tipo de langostino de la imagen.

Esta segunda fase se centra en la extracción y definición de dicho vector de características, que va a permitir concretar unos valores numéricos que definan el individuo estudiado. De acuerdo con lo visto sobre las diferentes especies de langostino con las que se están trabajando en el presente proyecto, se ha decidido tomar como características principales las tres siguientes tras un análisis de diferentes imágenes:

- a) Tamaño total de la muestra discretizada
- b) Color medio de la muestra discretizada
- c) Grado de rayado de la muestra discretizada

En esta fase se van a extraer las mismas y normalizarlas para obtener un vector de tres valores, que será el empleado en la tercera fase, de clasificación.

A continuación, se exponen los procedimientos y códigos finales desarrollados para realizar estas tareas, tras diferentes posibilidades.

a) Tamaño total de la muestra discretizada

Las premisas en las que se basa este procedimiento son, que las imágenes están tomadas todas a la misma distancia del objetivo (cámara fija) y con idéntica resolución.

Teniendo en cuenta ambas suposiciones, el código sumará el número de píxeles que son considerados como langostino tras la discretización final del apartado anterior, es decir, todos aquellos valores de la matriz RGB donde alguno o varios de estos valores de color sean diferentes del máximo.

La ventaja de emplear este método de área es que su resultado no depende de la forma en que la muestra se encuentre, sea estirado o encogido en forma de "c", dos posiciones habituales en estos animales.

Los tamaños habitualmente se miden en intervalos, donde cada intervalo es el número de ellos por unidad de masa. Por ejemplo, los que pertenezcan al intervalo 1 pueden ser aquellos de menor tamaño, con un calibre 40/50; los que pertenezcan al intervalo 4, aquellos de mayor tamaño, con un calibre de 10/20. Esto significaría que con un tamaño 1 se necesitarían entre 40 y 50 especímenes para sumar un kilogramo, y en el caso de tamaño 4 serían entre 10 y 20 los que sumarían dicha unidad de masa.

Debido a que esta suma puede realizarse a la vez que el cálculo del color medio en el langostino, el código se implementará de forma conjunta en el siguiente apartado.

En cuanto a la correspondencia de píxeles con los tamaños, se realiza de forma empírica a través de una serie de muestras cuyo tamaño es valor conocido, así como las condiciones en que es tomado (distancia focal de la cámara y resolución de la imagen), que se considerarán fijas en cualquier operación.

Una vez creada esta base de datos, el programa podrá comparar con el nuevo sujeto para conocer su característica de tamaño.

En función de la especie, los tamaños que se capturan tienen un mínimo y un máximo anatómicamente definidos, en los que se basa esta primera característica para su clasificación.

b) Color medio de la muestra discretizada

Se ha definido la primera de las características: el tamaño del elemento. Ahora se va a presentar la referida al color de su superficie.

Se supone que la iluminación del espécimen varía únicamente entre unos márgenes razonables. Esta iluminación constante será lo habitual en la mayoría de los casos, ya que la clasificación y obtención de imágenes se realiza bajo techo, iluminado por focos fijos, ya sea a bordo o en tierra.

El color medio de la muestra va a tener en cuenta todo el cuerpo, cabeza y cola. Además, dependiendo de la posición del espécimen y la discretización previa, puede que se cuenten también las patas, en mayor o menor medida según el caso. Por ello, en la primera fase, se ha trabajado para realizar una umbralización más ajustada con un método propio frente a los métodos prediseñados en las librerías de Matlab.

El color medio se obtendrá en forma vectorial [R G B], mediante el código que sigue:

```
function [colorM, area]=colorM(Im)
dimImg=size(Im);
cR=0;
cG=0;
cB=0;
nP=0;
Im=im2double(Im);
for i=1:dimImg(1)
    for j=1:dimImg(2)
        if Im(i,j,:)~=1 % Si es langostino
            cR=cR+Im(i,j,1);
            cG=cG+Im(i,j,2);
            cB=cB+Im(i,j,3);
            nP=nP+1;
        end
    end
end
colorM=round([cR cG cB]*255/nP);
area=nP/(dimImg(1)*dimImg(2));
```

De esta manera se obtiene el vector RGB: (llamado *colorM* en la función) con los valores de color medio en escala de 0 a 255 por un lado y, por otra parte, el valor área, donde el número de píxeles que pertenecen al langostino se dividen por el tamaño total de la imagen. Este cálculo permite seguir conociendo el tamaño del espécimen, aunque se cambie la resolución de la cámara que está tomando las imágenes, siempre y cuando la distancia focal no se vea modificada.

c) Grado de rayado de la muestra discretizada

La tercera y última característica elegida para identificar cada espécimen se basa en una cualidad física que presentan determinadas especies de langostino, el rayado.

Como se ha visto, las diferentes especies tienen una tonalidad o color diferente, en algunos casos más claramente diferenciado. Adicionalmente, pueden contar con diferencias físicas visuales consistentes en una variación del color en un área determinada, normalmente en forma de rayas sobre su dorso. Esa característica se convierte en un punto importante para discriminar, pues la existencia o no de este rayado permite descartar algunas especies y, en conjunto con las dos características previamente expuestas, pueden describir un espécimen basándose en diversas cualidades visualmente apreciables (y por tanto computables con visión artificial).

Para detectar la presencia del rayado en el cuerpo del animal se hacen dos posibles aproximaciones, a fin de encontrar la idónea para todos los casos.

- Aproximación por umbral

Se basa en la premisa de que la anomalía en el color por rayado se presenta como una región más oscura que el resto del cuerpo del espécimen. Empleando la imagen original ya umbralizada, se convierte a escala de grises, como en la primera fase, y se busca un valor umbral suficientemente bajo como para eliminar de la matriz resultante todos aquellos píxeles que no sean suficientemente oscuros como para pertenecer a dicho rayado.

Se empleará la función presentada en la fase inicial, que únicamente tiene en cuenta el valor de umbral dado para la discretización. Es importante definir ese valor para cada imagen, por lo que se aplicará un coeficiente reductor sobre el valor calculado, en este caso como el valor en escala de grises del color medio de la muestra.

- Aproximación por gradientes medios

Esta segunda propuesta requiere un cálculo adicional, ya que se basa en la diferencia de intensidad de dos puntos seguidos, es decir, en la derivada o gradiente orientado en una determinada dirección.

Al igual que en la aproximación por umbral, se tratará de destacar aquellos píxeles del langostino que puedan indicar la presencia del rayado, para comparar sobre la imagen anterior el número de ellos que han sido identificados como tal. El cociente de ambos será el porcentaje que interesa para poder basar la característica del vector en este valor.

Para esto es importante destacar el primer desafío, en cuanto a falso resultado que puede aparecer con alteraciones en la imagen que no sean debido al rayado: si se tienen en cuenta dos puntos contiguos que pertenezcan al espécimen, es muy posible que alteraciones diferentes de la buscada, como por ejemplo el punteado que algunos langostinos pueden presentar, sea tenido en cuenta sin pertenecer realmente a un conjunto de rayado, aunque si supongan alteraciones en la homogeneidad superficial.

Para evitar este problema se deberán tomar en cuenta no solamente el valor de dos píxeles contiguos para comprobar la tendencia (crecimiento o decrecimiento de valores), si no la media de los valores anteriores y posteriores al punto estudiado. De esta manera nos aseguramos de que la tendencia pertenece a la región y no a una anomalía aislada.

Debe tenerse en cuenta que del número de píxeles vecinos estudiados dependerá la mayor o menor sensibilidad del resultado. De esta manera, si de un mismo sujeto se toman imágenes a diferente distancias focales, el valor N de píxeles vecinos debería ser diferente para tener un resultado similar (menor a mayor distancia del objetivo).

Se ha decidido tomar el valor N como una fracción del número total de píxeles que pertenecen al propio animal, obteniéndose de forma empírica un valor aceptable (17).

$$N = 1 + 4 \cdot nP \cdot 10^{-5} \quad (17)$$

Donde nP es el número de píxeles que pertenecen al sujeto. Así el código autoajustará su precisión al tamaño del langostino inspeccionado.

El otro parámetro que se ha ajustado en la función es el valor límite a partir del cual esa diferencia en el gradiente se considera suficientemente grande como para ser tomada en cuenta. Debe ser reducido como para notar variaciones, aunque sean rayados poco definidos, pero no tan pequeño como para confundir zonas no rayadas con éstas. Valores entre un 6% y un 15% del umbral calculado por Otsu en fases anteriores dan resultados suficientemente buenos, por ello se toma como valor un 10% del umbral.

Tanto este valor límite como el de N son calculados en la función, que se muestra a continuación.

Se observa en la figura 50 dos langostinos diferentes evaluados con los dos métodos descritos, el primero pertenece a una clase de langostino sin rayado y el segundo a una clase con rayado. En todas las soluciones se muestra el porcentaje de rayado detectado sobre el animal con cada método.

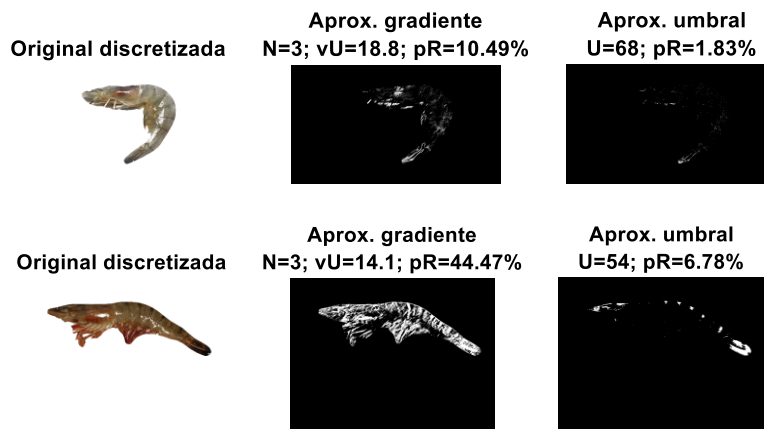


Figura 50. Comparativa extracción de rayado

```

function [Iray, porRay1]=rayas2(Im,U,sample, nP, area, Ip, pF)
Ia=rgb2gray(Im);
dimImg=size(Im);
Iray=zeros(dimImg(1),dimImg(2));
vU=0.1*U; % Valor límite del gradiente
dH=0;
dV=0;
nPr=0;
N=1+round(nP*4e-5); % N° de píxeles de la media para el gradiente
for i=(N+1):(dimImg(1)-(N+1))
    for j=(N+1):(dimImg(2)-(N+1))
        lf=0;
        rg=0;
        up=0;
        dw=0;
        for k=1:N
            lf=lf+Ia(i-N,j);
            rg=rg+Ia(i+N,j);
            up=up+Ia(i,j-N);
            dw=dw+Ia(i,j+N);
        end
        if Ia(i-1,j)~=255 && Ia(i+1,j)~=255 % Si es l. horizontal
            dH=abs(lf-rg);
        end
        if Ia(i,j+1)~=255 && Ia(i,j-1)~=255 % Si es l. vertical
            dV=abs(up-dw);
        end
        if dH>=vU || dV>=vU
            Iray(i,j)=1;
            dH=0;
            dV=0;
        end
    end
end
Ib=bordes(Ip,pF);
for i=1:dimImg(1)
    for j=1:dimImg(2)
        if Ib(i,j)==255 % Si es borde
            Iray(i,j)=0; % se elimina
        end
    end
end
for i=1:dimImg(1)
    for j=1:dimImg(2)
        if Ia(i,j)~=255 && Iray(i,j)==1 % Si es l. y "rayado"
            nPr=nPr+1;
        end
    end
end

porRay1=nPr/nP*10000; % Porcentaje de rayado sobre el total
porRay1=floor(porRay1)/100; % Con dos decimales

```

Siendo la función *bordes* una función auxiliar basada en el mapa de probabilidades para detectar los bordes y así poder suprimirlos del resultado. De otra forma se falsearían los datos al añadir en el rayado los bordes, que no pertenecen al espécimen.

Dicha función es la que sigue:

```
function Ib=bordes (Ip,pF)
dimImg=size (Ip);
Ib=zeros (dimImg (1),dimImg (2)); % Matriz de pertenencia
for i=1:(dimImg (1))
    for j=1:(dimImg (2))
        if Ip (i,j)>pF && Ip (i,j)<1.3*pF % Borde
            Ib (i,j)=255;
        else
            Ib (i,j)=0;
        end
    end
end
end
```

Tras realizar esta prueba sobre varias imágenes de diferentes especies y condiciones ambientales variadas, se descubre que los especímenes rayados tienen valores superiores al 30% y los no rayados inferiores, en términos generales.

3.2.3. TERCERA FASE: CLASIFICACIÓN

Tras una primera fase de identificación y discretización, se han extraído una serie de características de la imagen en la segunda fase.

Se muestra en la figura 51, el esquema de trabajo correspondiente a esta última fase.

El siguiente paso consiste en, a través de estas características, clasificar la especie de langostino entre las diferentes existentes en la base de datos. Para esta labor se emplean los vectores de características aplicados en clasificadores.

A continuación, se va a desarrollar el proceso de obtención y optimización del vector de características que se extraerá para cada espécimen analizado.

Posteriormente, se expondrán los diferentes clasificadores que se van a aplicar a dicho vector para tomar una decisión sobre el tipo de langostino al que el sujeto analizado tiene mayor probabilidad de pertenecer.

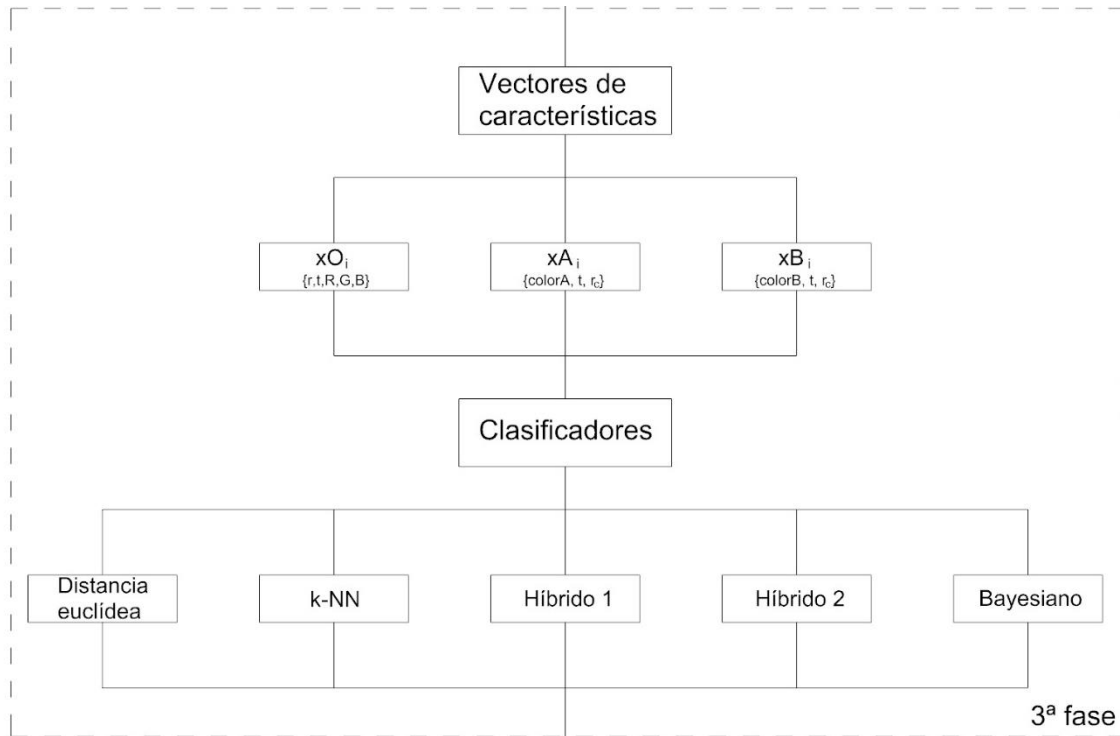


Figura 51. Esquema trabajo tercera fase

a) VECTOR DE CARACTERÍSTICAS

El primer paso que se realizará en esta tercera fase es una reducción y normalización de las variables, para poder trabajar con ellas.

Cuanto menor número de características se tengan, más eficiente será el programa, siempre y cuando sean suficientes para una correcta clasificación. Por ello, aunque se tienen tres características de la segunda fase, la referente al color RGB es, en realidad, una característica con tres valores, lo que significaría que el vector características sería un vector de cinco valores:

$$x_i = \{R, G, B, t, r\}$$

Siendo t, el tamaño y r, el rayado.

Uno de los puntos más importantes del proyecto presentado, aparte de la novedad que supone en el estado del arte sobre la clasificación del langostino de forma automatizada, consiste en la aplicación y definición de un vector de características concreto y eficiente para aplicar en el clasificador elegido.

Se cuenta con una base de datos de cinco clases de langostino, y un total de 50 fotos por especie, como se muestra en el anexo 7.1.

A continuación, se va a estudiar cómo procesar estas características para simplificarlas en la medida de lo posible, maximizando al mismo tiempo las diferencias entre los distintos clústeres/agrupaciones de datos.

- **Tamaño**

El tamaño es lo que dará una información menos característica para la clasificación por tipo, aunque será de ayuda para inclinar la balanza hacia un tipo u otro en caso de duda. Tiene sin embargo una finalidad doble: por un lado, aporta un apoyo en la clasificación por especie y, por otra parte, servirá para la posterior clasificación por tamaño.

Es fácil observar sobre la base de datos (en base al valor promedio de esta característica), que este número influye en la clasificación, en particular cuando el langostino pertenece a la clase más grande, pero en el resto de los tipos su relevancia es menos notable (tabla 1).

Tabla 1. Tamaño medio por especie BBDD

Tamaño medio	
PK	2.62
PM	2.19
PP	3.28
PV	1.67
J	10.51

Esta variable se toma como el porcentaje de la imagen que el espécimen está ocupando, y podrá tomar valores entre 0 y 100.

- **Rayado**

En el procesado de la imagen se ha aplicado una máscara basada en el cálculo de las medias de la primera derivada para obtener el grado de rayado. De esta manera se ha obtenido una cifra, expresada en porcentaje sobre el total del espécimen (a fin de no verse afectado el valor por el tamaño del mismo), que indica el grado de anomalías o variaciones de color sobre la superficie del animal.

En el mismo procesado se eliminan los bordes para que no sumen en esta característica falseando el resultado. En cualquier caso, existen pequeñas variaciones que el programa tomará como rayado sin serlo, por lo que resulta conveniente darle notoriedad a aquellas imágenes donde el grado de rayado sea notablemente superior, y restar importancia a aquellos donde sea algo menor, con el objetivo de agrupar en mayor medida los resultados obtenidos.

Se puede observar, en la figura 52, comparando el tamaño y el porcentaje de rayado directamente, que los clústeres en la dimensión del porcentaje de rayado se extienden hasta 20, en una agrupación algo tosca. A fin de ayudar en esta agrupación de nubes de puntos, sobre todo en reducir su espacio horizontal, se ha considerado la posibilidad de aplicar una corrección en los mismos basada en la función exponencial, de la forma siguiente (18):

$$r_c = e^{\frac{r}{k}} \quad (18)$$

Siendo r_c , el valor de rayado corregido; r , el valor directamente obtenido, y k , una constante determinada, que tras varias iteraciones y posibilidades se ha definido en $k=25$. La forma de agrupación es prácticamente igual, pero los valores de las abscisas han variado, reduciendo las distancias entre clústeres vecinos (figura 53).

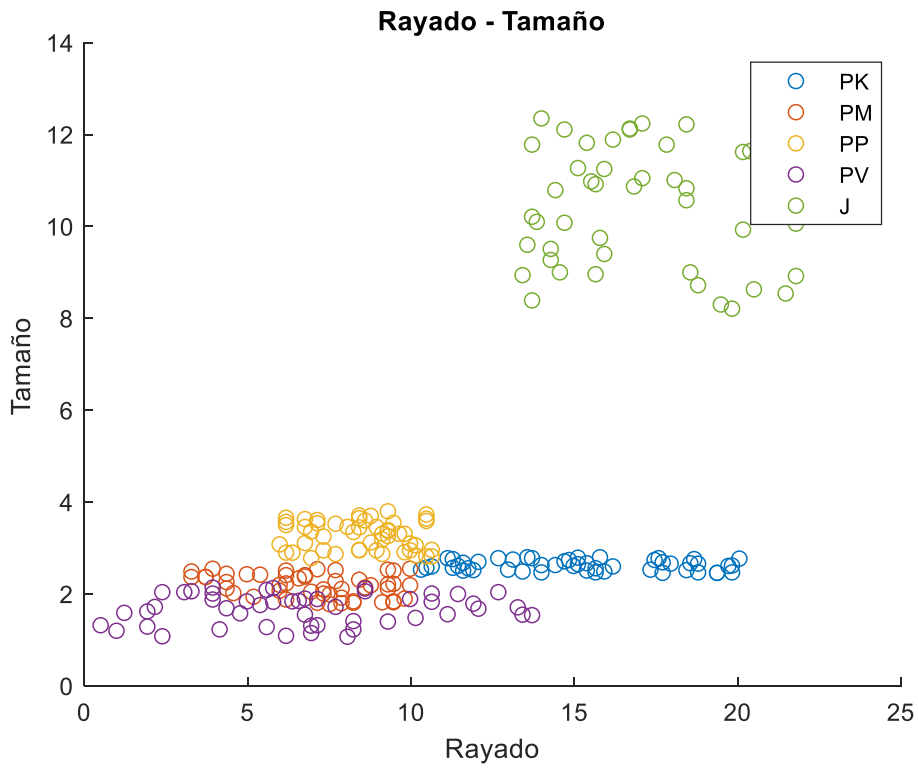


Figura 52. Distribución bidimensional rayado-tamaño

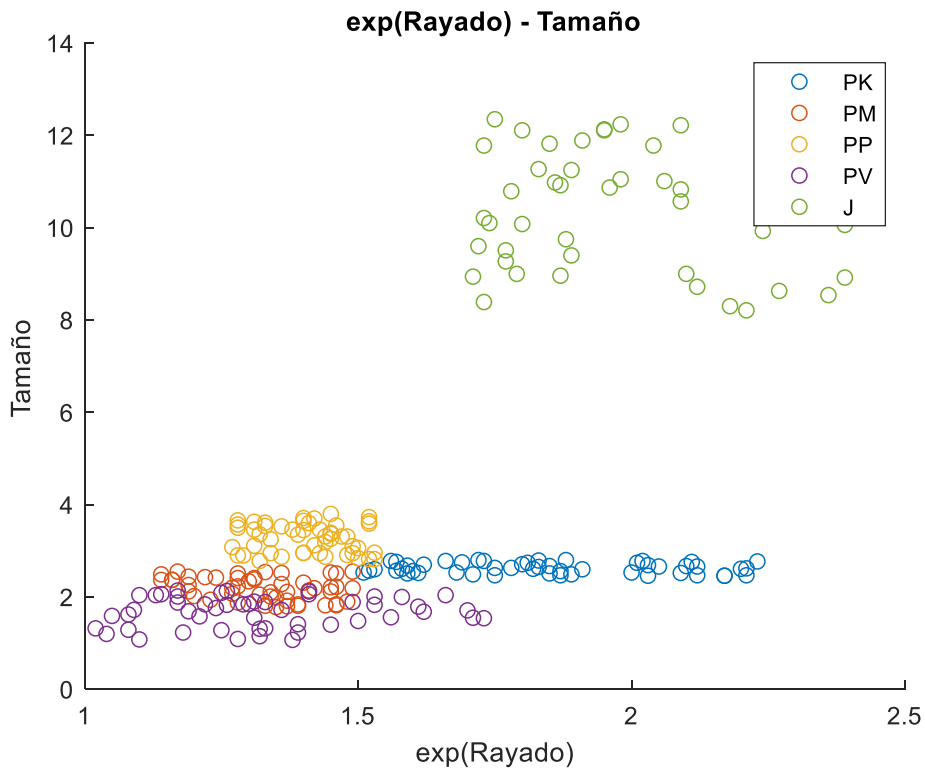


Figura 53. Distribución bidimensional exp(rayado)-tamaño

- Color

La tercera característica, el color, viene representado por tres valores en cada imagen, entre 0 y 255 para el rojo (R), verde (G) y azul (B). En la tabla 2 se muestra, para cada tipo de langostino, el valor promedio de cada parámetro de color de todas las muestras.

Tabla 2. Color medio por especie BBDD

Color medio			
Tipo	R	G	B
PK	88	45	30
PM	72	43	30
PP	93	49	29
PV	85	62	51
J	49	37	32

La primera conclusión evidente es que el factor más diferenciador entre especies parece ser el rojo, y el menos importante el azul, siendo que el verde tampoco aporta una diferencia notable, a excepción en ambos casos de la especie PV, cuyos valores de G y B resultan algo más elevados que la media de las otras especies.

Estas deducciones lógicas en base a los resultados deben ser probadas, para encontrar de manera eficiente una solución razonable a la convergencia de la característica color (RGB).

Escala de grises:

Un sistema clásico que relaciona entre sí los diferentes colores, dándoles pesos no homogéneos, es la conversión a escala de grises NTSC que ya se empleó en la fase de discretización del presente proyecto, y que responde a la siguiente ecuación (16)

De esta manera obtendremos un vector de características x_{A_i} definido de la siguiente forma:

$$x_{A_i} = \{ColorA, t, r_c\} = \left\{ 0.2989 \cdot R + 0.580 \cdot G + 0.1140 \cdot B, t, e^{\frac{r}{k}} \right\}$$

Y cuya representación espacial, tanto bidimensional como tridimensional, incluyendo los centros de los clústeres, se muestra en las figuras 54, 55 y 56, donde se observa una superposición entre elementos del grupo PP con PV y PM especialmente, y puede ocasionar una clasificación poco acertada en ese caso.

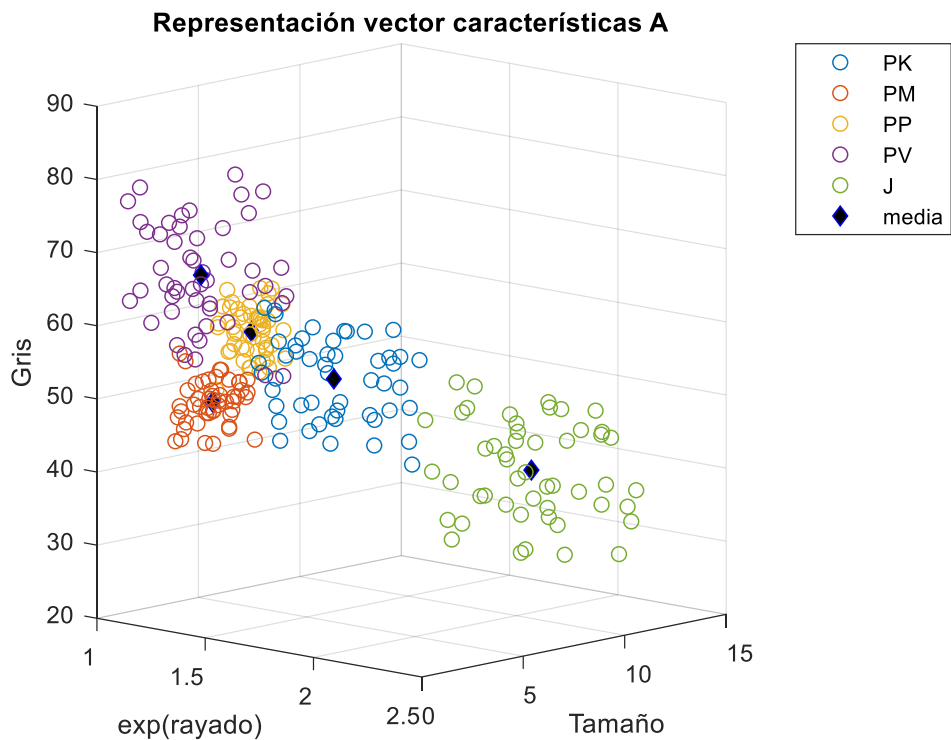


Figura 54. Representación 3D vector de características A

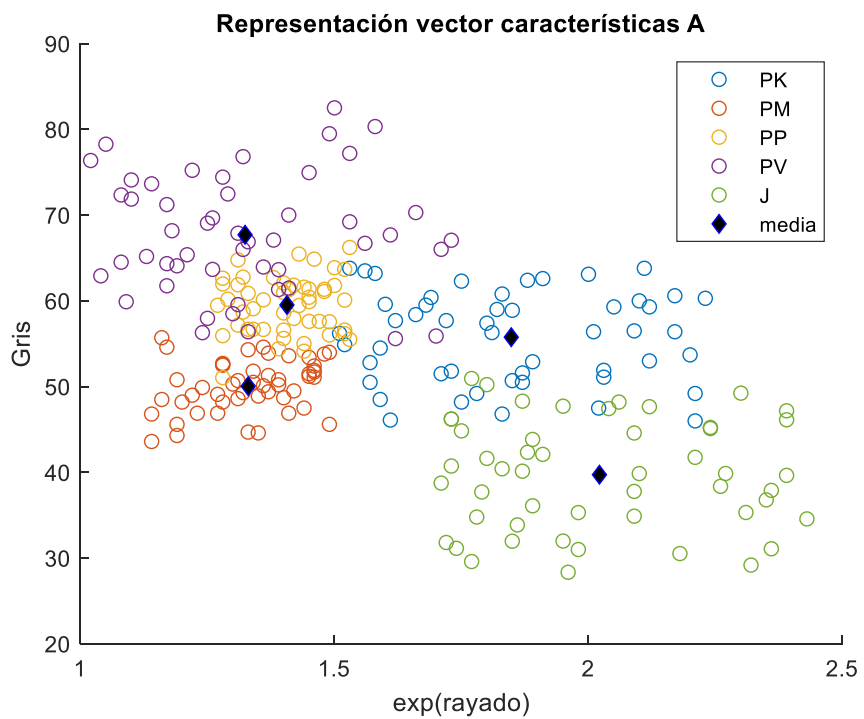


Figura 55. Representación 2D gris-exp(rayado) vector de características A

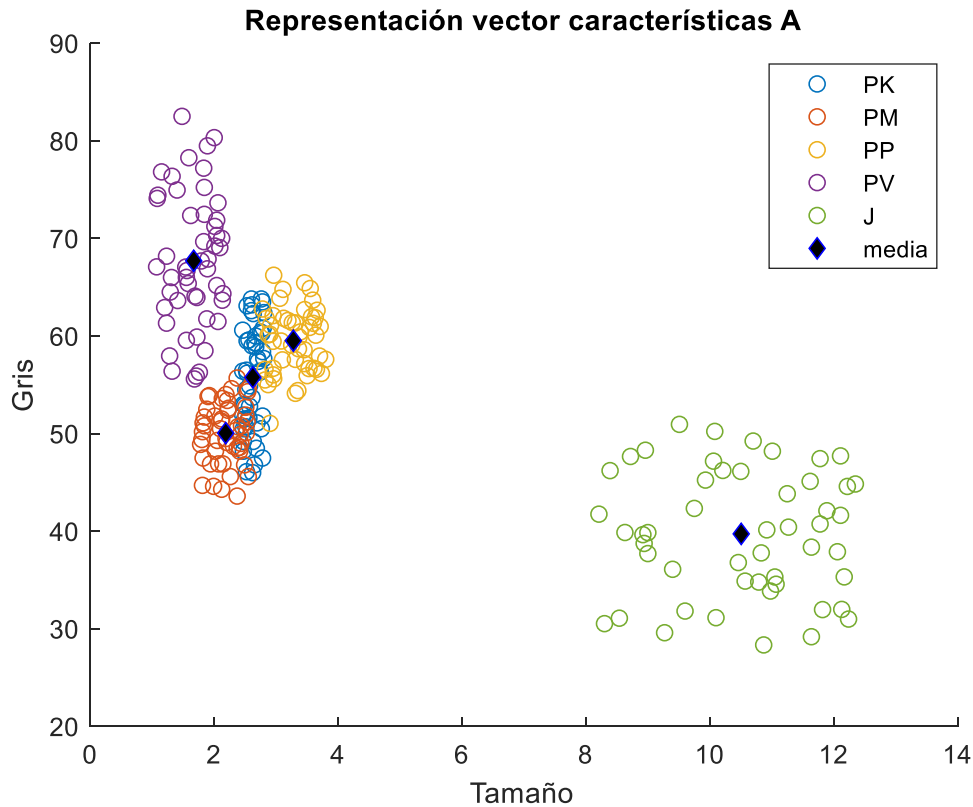


Figura 56. Representación 2D gris-tamaño vector de características A

Relación R, G, B

La opción mostrada mediante escala de grises tiene algunas divergencias en la agrupación que pueden dificultar la clasificación. Por ello va a tratarse de obtener una relación original entre las componentes (R, G, B) que optimice el resultado, basándose en los valores de la base de datos.

Si comparamos entre si las diferentes componentes del color podemos ver, como se muestra en las figuras 57 y 58, que el color rojo claramente es determinante en la clasificación del color, ya que diferencia entre si los diferentes clústeres, al menos en tres grupos: el grupo más cerca del origen de coordenadas perteneciente a los langostinos tipo J, el segundo grupo donde se encuentran los langostinos tipo PM y una parte menor de PV y, por último, el tercer grupo, más alejado del origen de coordenadas donde PP, PK y gran parte de PV confluyen.

De esta forma el rojo, aun siendo el factor más influyente del color, no parece suficiente por sí mismo para la clasificación que se persigue.

Por otra parte, la comparativa entre G y B muestra una proporcionalidad importante, lo que se traduce como una reducida ayuda en la clasificación del espécimen (figura 59).

Finalmente, en esta misma figura se destaca la proporcionalidad entre el rojo y los otros colores: mientras que el rojo crece proporcionalmente con el verde, siguiendo una clara linealidad, el azul y el rojo se distribuyen en regiones “acotadas”, tanto vertical como horizontalmente.

La relación que se va a buscar será del parámetro R en función de G y/o B, desestimando la relación entre G y B por carecer de interés real para el propósito del proyecto. Asimismo, se debe procurar una solución que maximice la diferenciación entre PK y PP para evitar que se solapen en la medida de lo posible.

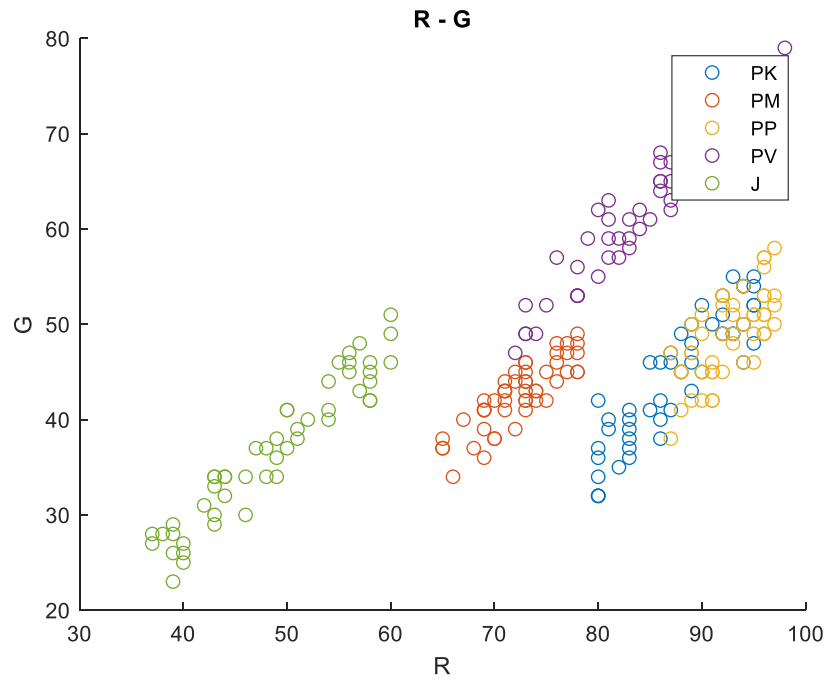


Figura 57. Representación 2D R-G

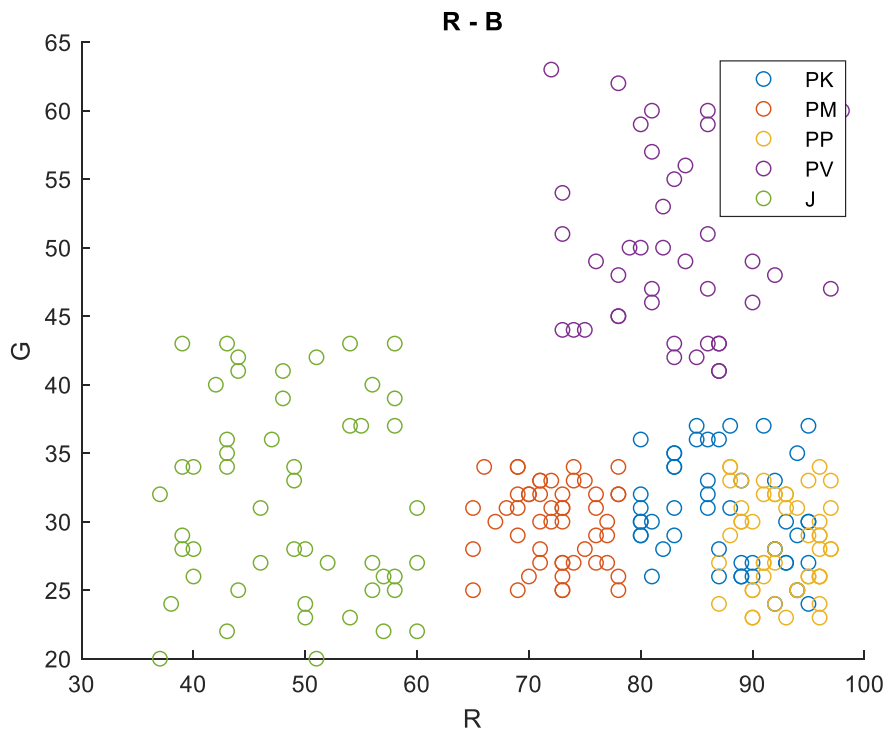


Figura 58. Representación 2D R-B

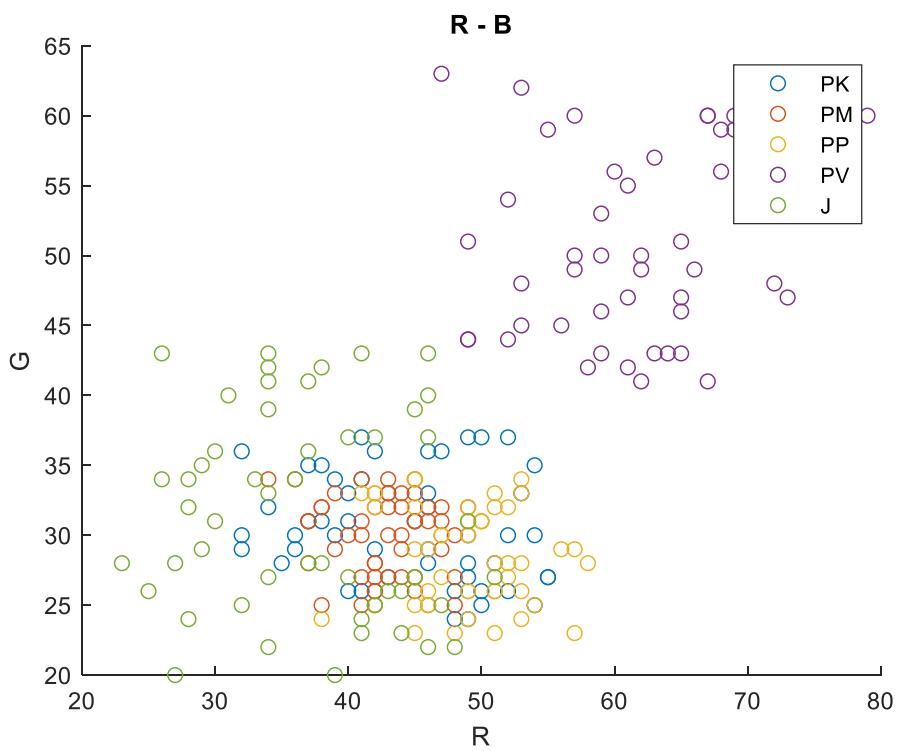


Figura 59. Representación 2D G-B

Las primeras aproximaciones serán las más simples o evidentes, tomando el rayado exponencial como eje de ordenadas para la comparativa de datos basados en color. Se muestran en las figuras 60 a 65 las distribuciones bidimensionales frente al rayado de: la intensidad de rojo R, la intensidad de verde G, las relaciones proporcionales entre R y G, y entre R y B, así como la diferencia de R y G, y la diferencia de R y B.

La representación del rayado frente a B se ha desestimado, de acuerdo a lo expuesto.

En estos casos se muestra la mayor diferenciación entre clústeres para el caso de cociente de rojo y azul, ya que en el resto de los casos los tipos PP, PM y PV tienen una distribución espacial solapada en general.

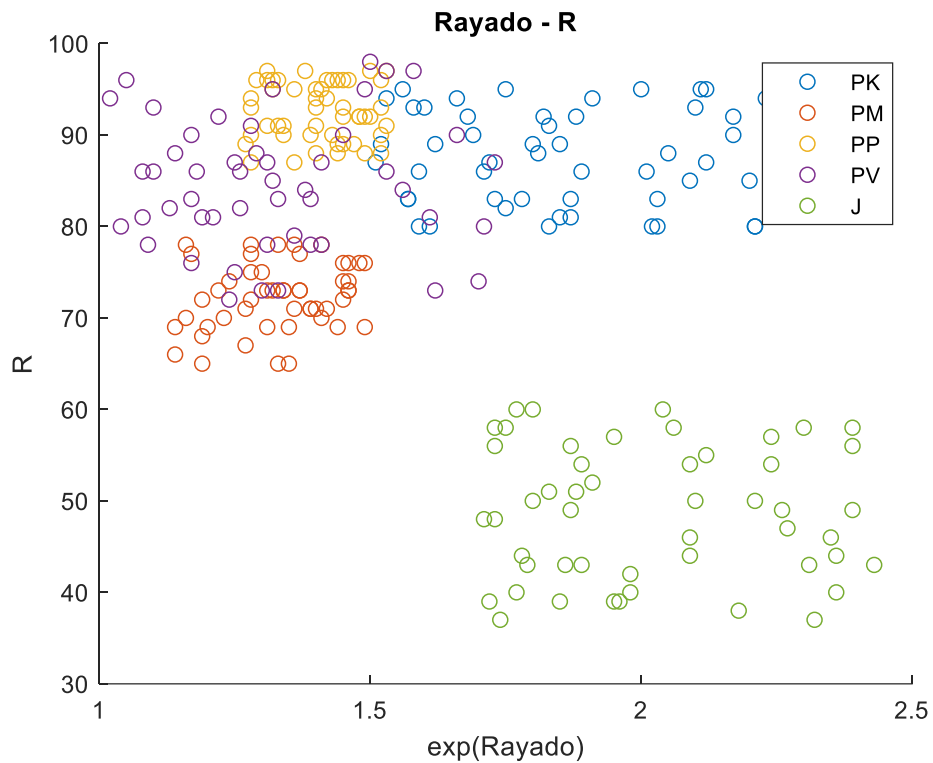


Figura 60. Representación 2D Rayado-R

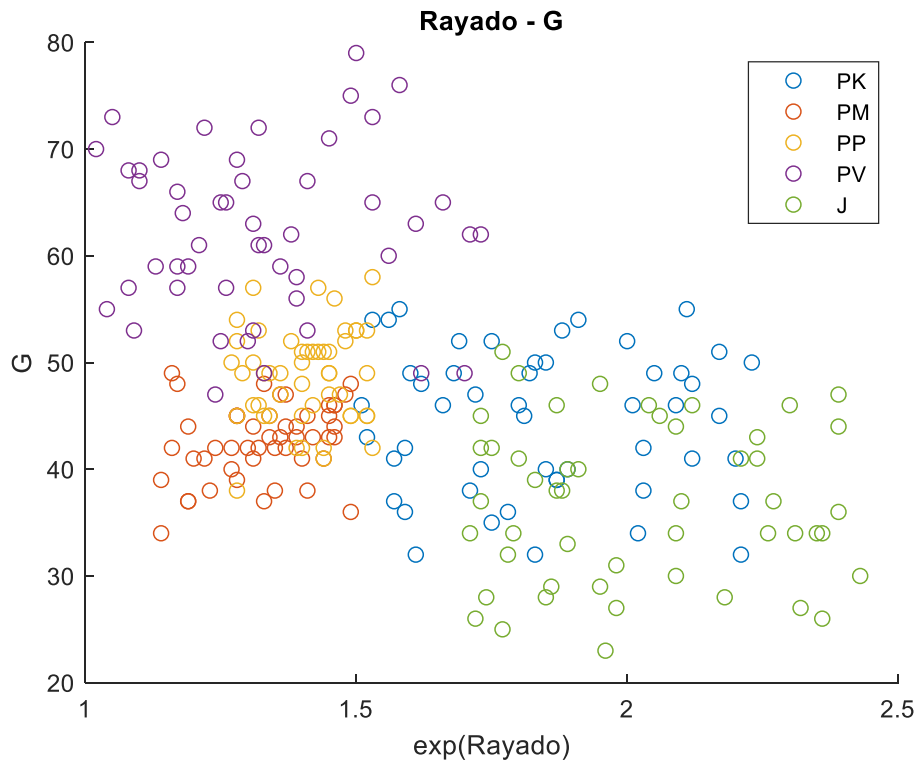


Figura 61. Representación 2D Rayado-G

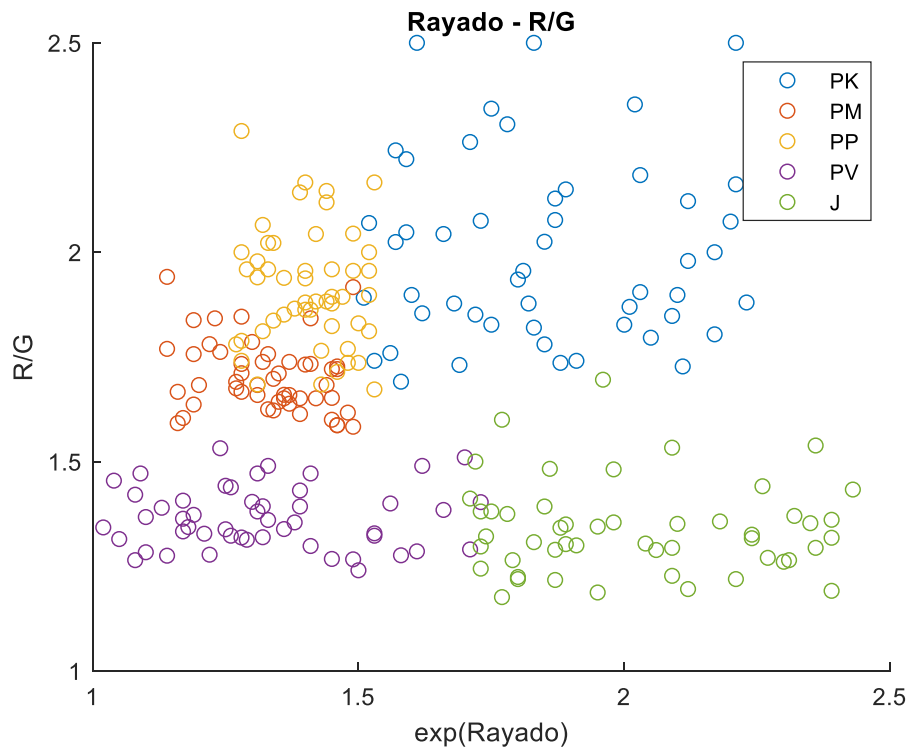


Figura 62. Representación 2D Rayado-R/G

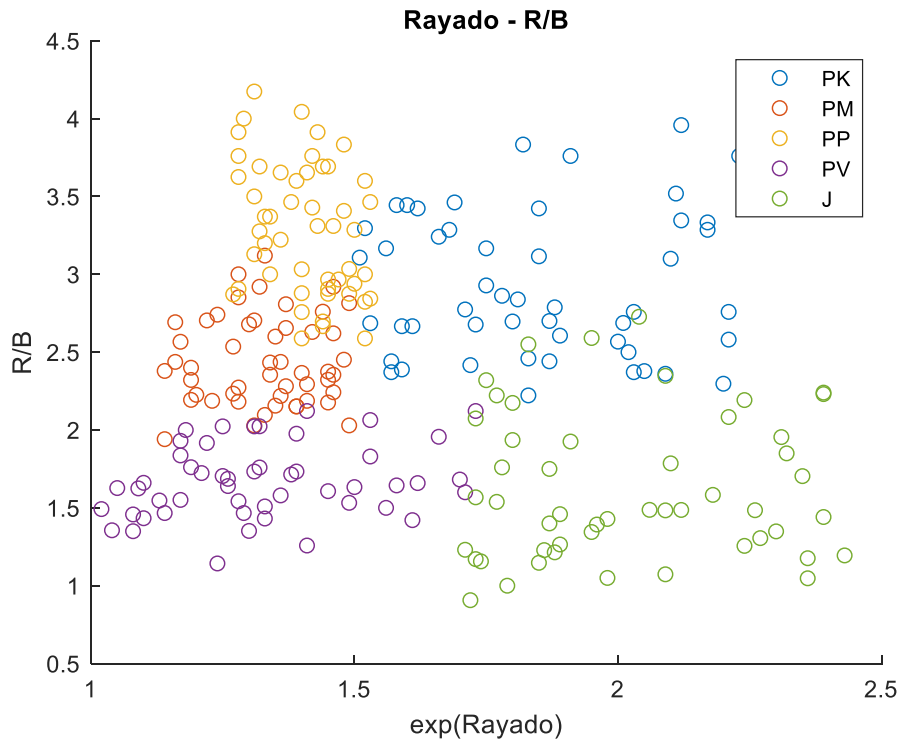


Figura 63. Representación 2D Rayado-R/B

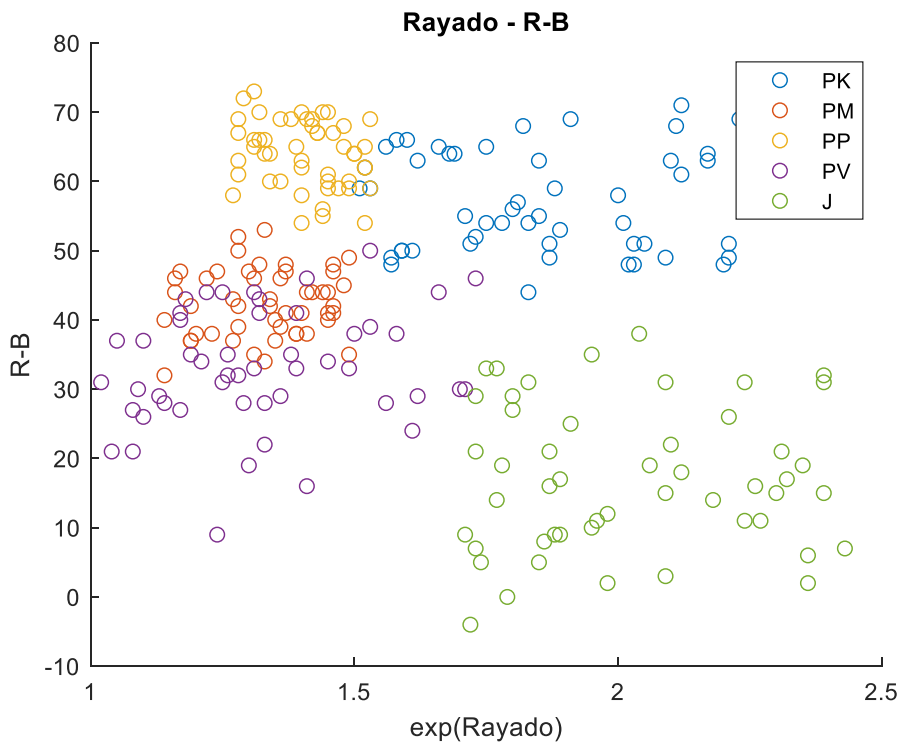


Figura 64. Representación 2D Rayado-(R-B)

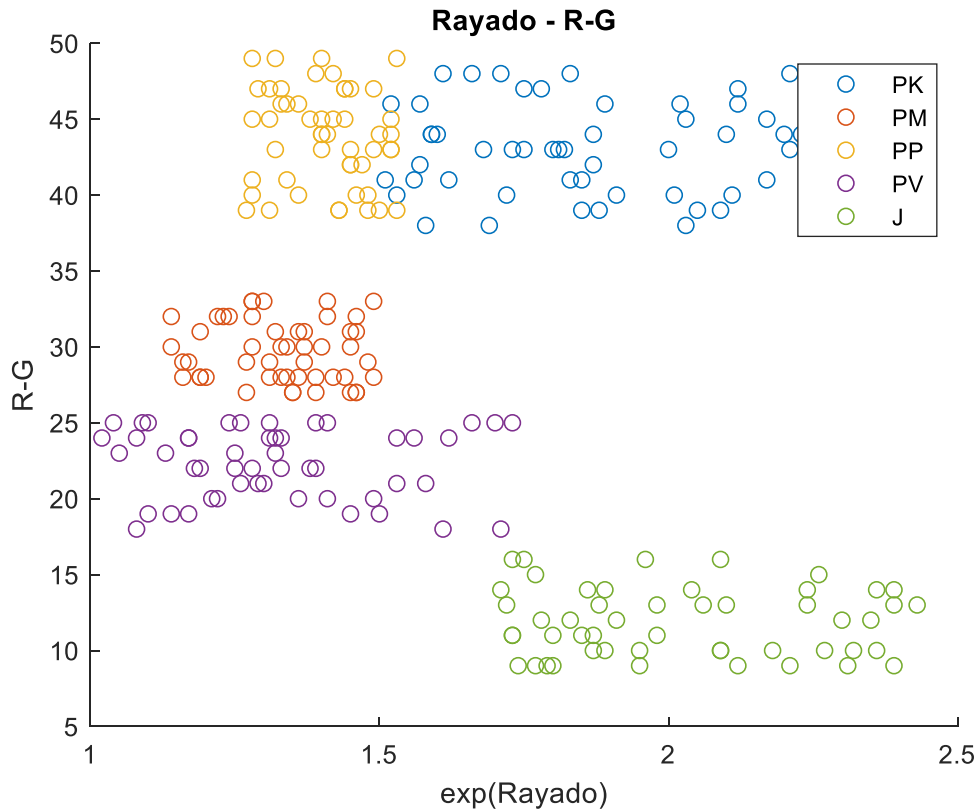


Figura 65. Representación 2D Rayado-(R-G)

Siguiendo el mismo razonamiento, parece lógico buscar un valor que relacione el caso anterior (R/B), con el valor de G de alguna manera, ya que así se tendrían en cuenta los tres parámetros del color (R, G, B). Para ello, se va a multiplicar dicho valor por la diferencia de rojo y verde, de forma que se le da un mayor peso al parámetro R, que ha mostrado una influencia más elevada en la diferenciación, y así se puede maximizar en todo caso la distancia entre clústeres o agrupaciones de datos.

Podemos definir el parámetro del color de la siguiente forma (19):

$$ColorB = (R - G) \cdot \frac{R}{B} \quad (19)$$

Esta es una propuesta original del autor para buscar la optimización del resultado, aunque es solo una de las posibilidades que pueden emplearse.

Para comprobar la efectividad de ambos sistemas van a compararse con los diferentes clasificadores, por lo que tendremos dos vectores de características a evaluar:

$$xB_i = \{ColorB, t, r_c\} = \left\{ (R - G) \frac{R}{B}, t, e^{\bar{k}r} \right\}$$

De esta manera se tiene todo lo necesario para obtener resultados de clasificación por especie, al introducir el vector de características en los clasificadores. Adicionalmente, el programa también podrá proceder a su clasificación por tamaño dentro de la especie.

Se puede observar en la figura 66 la representación tridimensional del resultado del vector sobre la base de datos, incluyendo los valores medios de cada clúster, si se compara con la distribución observada para el vector anterior, puede verse una mayor separación entre clases, especialmente atendiendo al grupo PP. En las figuras 67 y 68 se muestran las proyecciones bidimensionales para ver de forma más cómoda el mismo hecho.

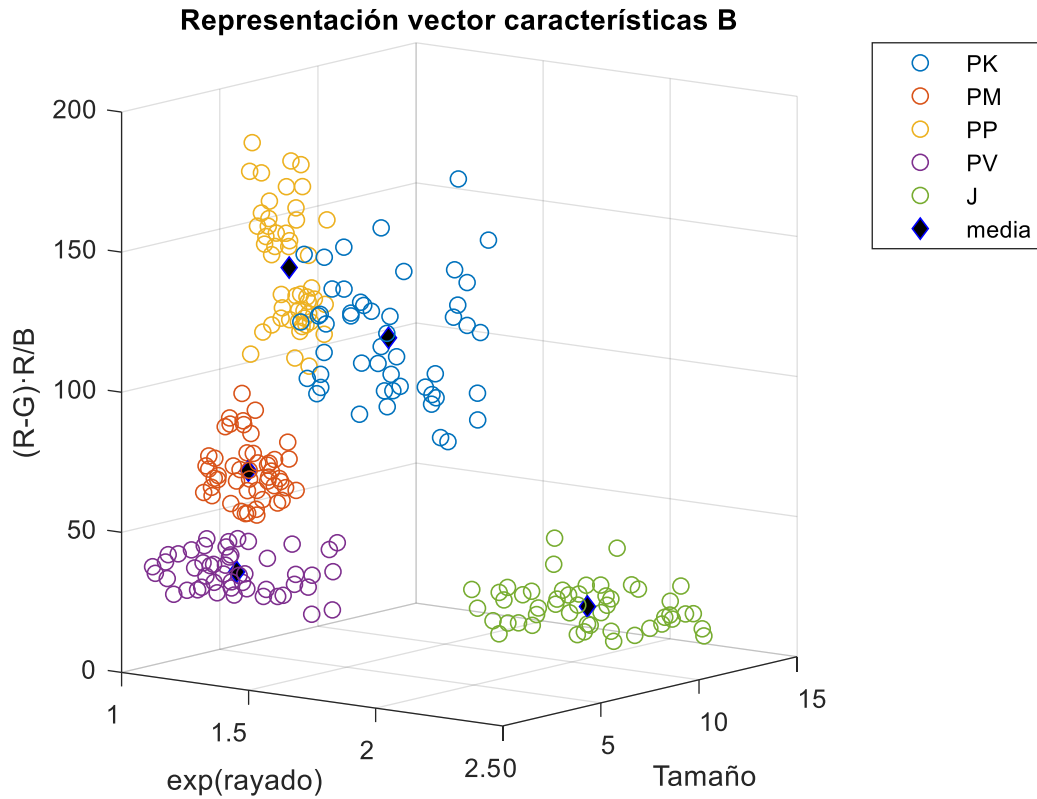


Figura 66. Representación 3D vector de características B

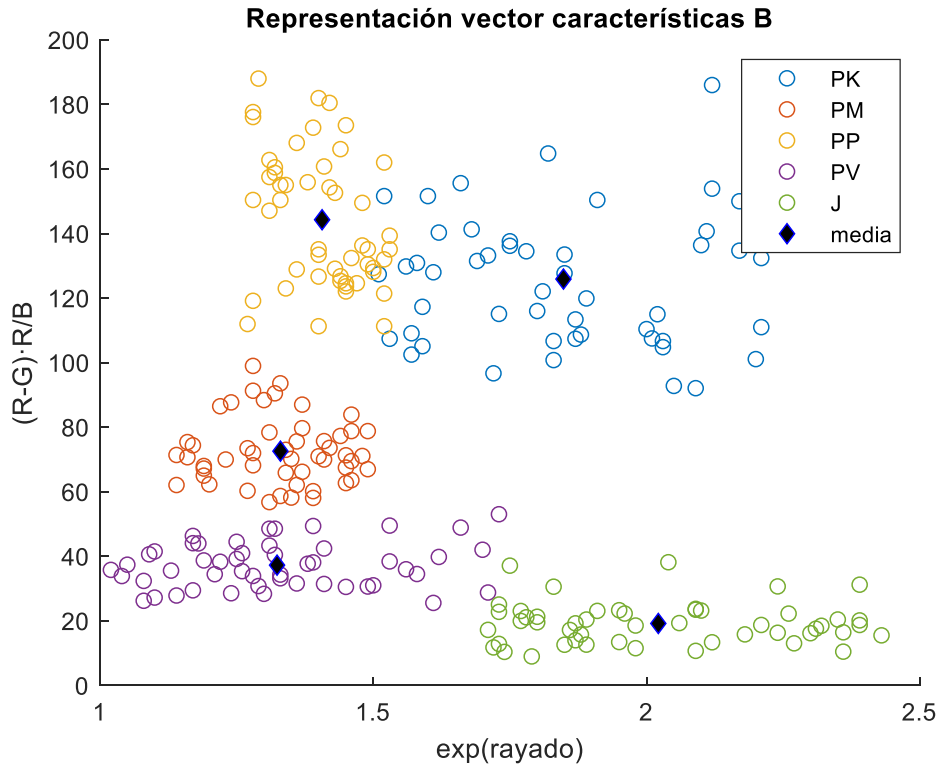


Figura 67. Representación 2D color- $\exp(\text{rayado})$ vector de características B

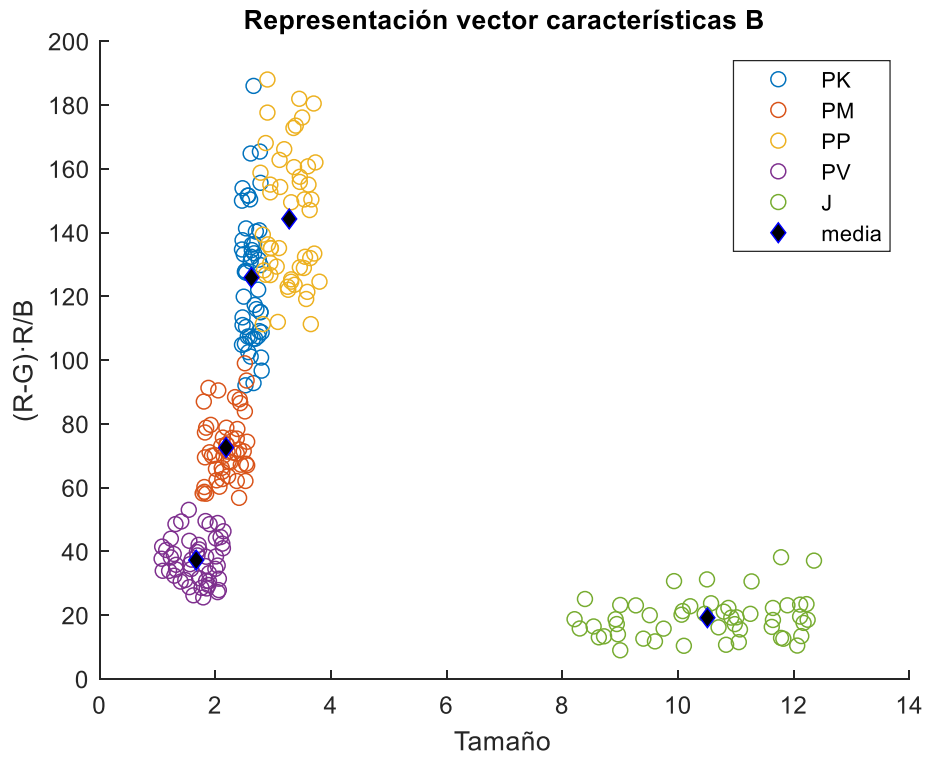


Figura 68. Representación 2D color-tamaño vector de características B

b) CLASIFICADOR

Existen diversas propuestas para tratar el vector de características y obtener a partir de él un resultado. Para ello se emplean los llamados clasificadores, que son sistemas de operaciones matemáticas entre el vector obtenido de la imagen a clasificar y la base de datos.

Se van a exponer brevemente algunas formas de clasificación que posteriormente se emplearán en la sección de resultados, aplicando los dos vectores de características definidos, a fin de encontrar el sistema más eficiente de clasificación de especies [24].

- **Clasificador por pertenencia binaria a clúster (Voronoi)**

Un sistema muy rápido por el reducido número de cálculos a realizar, aunque limitado en sus conclusiones.

La definición binaria se refiere a que en sus resultados no hay solapamiento ni matiz, el langostino pertenecerá únicamente a un tipo, y no pertenecerá a los demás.

El programa calculará la distancia entre el vector de características y los promedios de cada clúster, para lo cual puede emplearse el método de cálculo de distancia que se prefiera, siendo el más habitual el de la distancia euclídea (20).

$$d_e = \sqrt{\sum_{i=1}^n (x_i - \bar{x}_i)^2} \quad (20)$$

Siendo:

- n: el número de características
- $x = \{x_1, x_2, \dots, x_n\}$: el vector de características
- $\bar{x} = \{\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n\}$: el vector de características medias para cada clúster

De esta forma, una vez se tiene la distancia entre el vector de características y los vectores pertenecientes a la media de cada clúster, aquella que sea menor será la que defina el grupo al que pertenece la muestra de la imagen. Esto es, pertenecerá al tipo de langostino de la región n-dimensional en que se encuentre.

Las fronteras de las regiones serán definidas por las coordenadas equidistantes a los centros de sus clústeres vecinos, y se forman así una serie de regiones conocidas como diagramas de Voronoi.

Este sistema resulta excesivamente arbitrario y, aunque sus fronteras se modifican ligeramente con los nuevos valores que se adquieren, su adaptabilidad es reducida. Además, como se observa en las cinco imágenes diferentes presentes en la figura 69 (vistas proyectadas de dichas separaciones sobre los valores de la base de datos), con este método habría un gran número de individuos (de la base de datos) que no serían clasificados correctamente al encontrarse estrictamente fuera de la región a la que pertenecen.

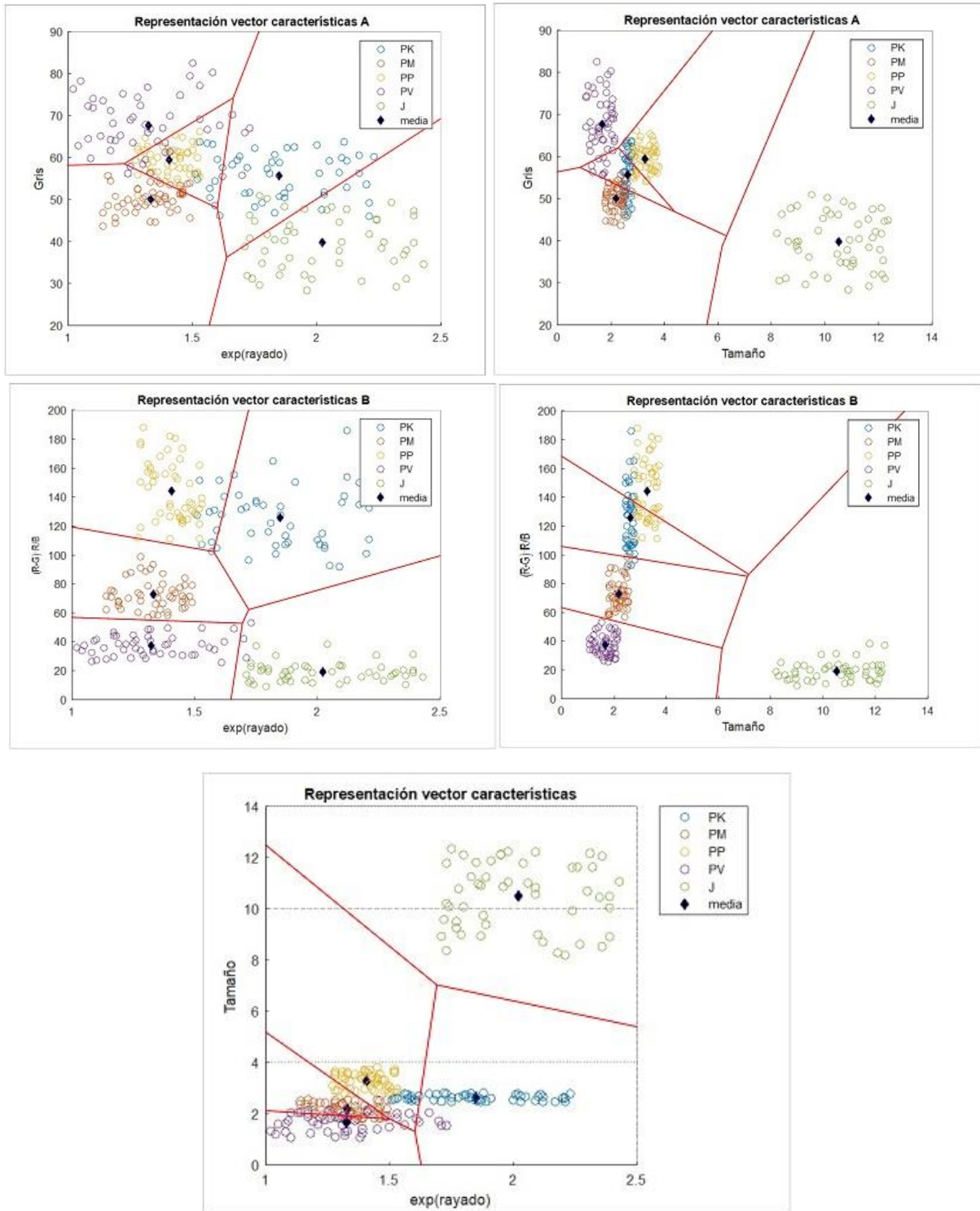


Figura 69. Diagramas de Voronoi bidimensionales

- **Clasificador por vecinos más cercanos (k-NN)**

Otro de los clasificadores más simples en cuanto a implementación se refiere: por vecinos más cercanos (nearest neighbours en inglés).

En este caso se tienen en cuenta los k vectores más cercanos en el espacio n-dimensional respecto al vector de características dado, basándose en la distancia elegida que, al igual que con el clasificador anterior, será la euclídea.

De esta forma el programa deberá encontrar las distancias euclídeas al resto de vectores de la base de datos, tomando como resultado el número de ellas especificado (k), con los más cercanos al vector de características de estudio.

Si tomamos por ejemplo una ampliación de la base de datos, representando bidimensionalmente el rayado exponencial frente a la característica de color definida en el método B, podemos ver un nuevo resultado, representado como una estrella roja (figura 70).

Si se toma k=3, el espécimen pertenecería a la clase PP, al tener dos representantes de ésta, frente a un único representante de PK. Si tomamos en cambio k=5, la clase del sujeto cambiaría a PK, ya que se añadirían 2 elementos perteneciente a PK, y ninguno de la clase PP.

Esto manifiesta la importancia del parámetro k, así como las limitaciones de este sistema de clasificación.

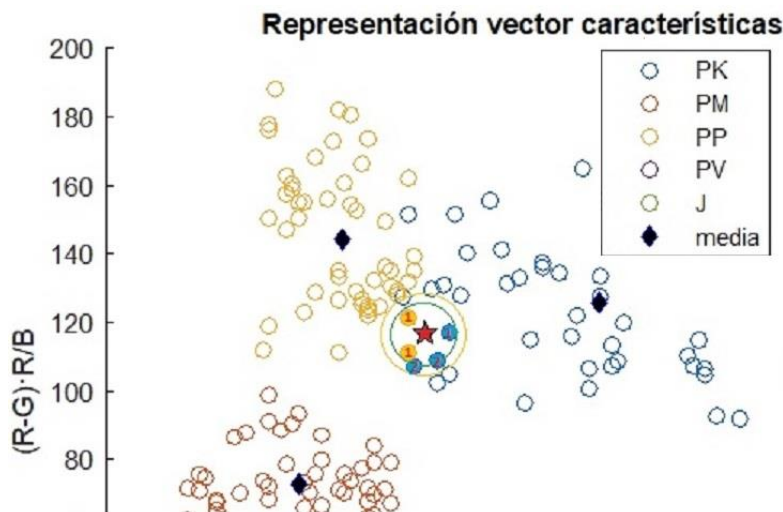


Figura 70. Representación bidimensional método k-NN

- **Clasificador híbrido k-NN y pertenencia a clúster (no binario)**

Los dos clasificadores mostrados tienen en común su poca adaptabilidad o, dicho de otra forma, cierta ambigüedad. Como respuesta a este hecho se define un clasificador intermedio, que tenga en cuenta el grado de pertenencia a un clúster junto con el número de vectores vecinos que pertenezcan al mismo, este sistema consiste en una combinación de los dos métodos anteriores, de tal forma que el programa deberá calcular dos cosas para poder decidir a qué conjunto pertenece la muestra:

- El número de k vecinos definidos, de idéntica manera al clasificador k-NN, que siguiendo el ejemplo anterior (figura 67) serían 2 PP y 1 PK para $k=3$, o bien 2 PP y 3 PK para $k=5$.
- La distancia entre el vector de características del sujeto y los centros de los clústeres a los que pertenecen los k vecinos definidos. En el ejemplo, las distancias euclídeas a los vectores medios de PP y PK, esta distancia se calcula como se mostró en el clasificador por pertenencia binaria ecuación (20).

Ahora, sin embargo, se pretende normalizar estas distancias antes de aplicarlas al problema. Para ello se divide la distancia euclídea d_e que se desea obtener por el sumatorio de distancias euclídeas a los centros de los clústeres implicados con el método k-NN. De forma general (21):

$$d_{eN} = \frac{\sqrt{\sum_{i=1}^n (x_i - \bar{x}_i)^2}}{\sum_{j=1}^m \sqrt{\sum_{i=1}^n (x_{ij} - \bar{x}_{ij})^2}} \quad (21)$$

Siendo:

- m : el número de clústeres implicados en el método k-NN
- $x = \{x_1, x_2, \dots, x_n\}$: el vector de características
- $\bar{x} = \{\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n\}$: el vector de características medias
- j : subíndice para indicar las diferentes clases

Para encontrar la pertenencia a cada clase se procedería al producto del número de vectores vecinos encontrados de una determinada clase j mediante k-NN por la unidad menos la distancia euclídea normalizada al centro del clúster de dicha clase j (22).

$$P(\text{pertenencia})_j = k_j \cdot (1 - d_{eN_j}) \quad (22)$$

- **Clasificador híbrido k-NN y pertenencia a clúster (no binario) alternativo**

Se pueden mejorar los resultados en determinados casos dándole más importancia a la distancia frente al número de vecinos. Una posible alternativa es la que sigue (23).

$$P(\text{pertenencia})_j = \frac{k_j}{10 \cdot d_{eN_j}} \quad (23)$$

- Clasificador Bayesiano

Este clasificador se basa en el modelo de distribución probabilística de Bayes, y es uno de los métodos más empleados por el elevado acierto de sus resultados para gran variedad de casos y situaciones. Existen además diferentes aproximaciones y variaciones del método.

El sistema, al igual que los anteriores, parte de una base de datos de vectores características, y de una muestra con su propio vector de características que será el que se evalúe. El objetivo del método es encontrar las diferentes probabilidades de pertenencia a cada uno de los clústeres a partir de las probabilidades compuestas de pertenencia a una clase determinada, en función de los valores del vector característico.

La formulación mediante el teorema de Bayes es la que sigue (24):

$$P(C|x_1, x_2, \dots, x_n) = \frac{P(C) \cdot P(x_1, x_2, \dots, x_n|C)}{P(x_1, x_2, \dots, x_n)} \quad (24)$$

Siendo:

- C: la clase a la que se busca su grado de pertenencia
- $x = \{x_1, x_2, \dots, x_n\}$: el vector de características

El numerador es únicamente una forma de normalización del resultado, y será el mismo para todas las probabilidades conjuntas de una muestra determinada.

Como el número de muestras es suficientemente grande y se espera crecimiento de éste con las identificaciones sucesivas que realice el programa, se va a considerar una distribución normal o gaussiana del espacio de características.

Este tipo de distribución tiene una densidad de probabilidad que responde a la formulación (25):

$$P(x_i|C) = \frac{1}{\sqrt{2\pi\sigma_i^2}} e^{-\frac{(x_i - \mu_i)^2}{2\sigma_i^2}} \quad (25)$$

Siendo:

- x_i : la característica a evaluar
- σ_i^2 : la varianza de x_i asociada a la clase C
- μ_i : el promedio o valor esperado de x_i asociada a la clase C

Por último, aunque en la realidad las probabilidades de encontrar una especie u otra dependen de características ambientales, época del año y otros factores externos, no existe un estudio al respecto en el cual basarse, por este motivo se va a tomar, considerándose correcto para este caso de estudio, una distribución equiprobable de las especies, que siendo cinco tendrá un valor de un 20%: $P(\text{especie})=0,2$.

4. ANÁLISIS Y DISCUSIÓN DE RESULTADOS

En base al proceso expuesto se va a proceder al análisis de 50 imágenes tomadas en condiciones ambientales similares y controladas de las cinco especies, de forma que el programa las procese de una en una, con el orden aleatorio en que le van llegando. Dicho orden, así como las características extraídas de cada espécimen y los cálculos efectuados sobre estas características, se muestran en la tabla 3, donde los colores indican de forma general el éxito, siendo el verde de mayor intensidad cuanto mayor es la tasa de acierto y la mayor intensidad de rojo a menor tasa de acierto.

Para la elaboración de estas tablas se ha definido el acierto o el fallo mediante la valoración de un experto, a fin de conocer el porcentaje de éxito que tiene el programa desarrollado y su posible operación autónoma (no supervisada).

Los resultados obtenidos se dividen en tres grupos que van a compararse, con cinco subgrupos cada uno de ellos. Los grupos generales se definen en base al vector de características seleccionado:

- Vector *bruto*: Con las cinco características originales directamente obtenidas de la imagen.

$$xO_i = \{Ray, t, R, G, B\}$$

- Vector *gris*: Con tres características, incluyendo el exponencial del rayado y la conversión a valor en escala de grises NTSC del RGB extraído de la imagen.

$$xA_i = \{ColorA, t, r_c\} = \{0.2989 \cdot R + 0.580 \cdot G + 0.1140 \cdot B, t, e^{\frac{r}{k}}\}$$

- Vector *RGRB*: Con tres características, incluyendo el exponencial del rayado y la conversión a valor propio del RGB extraído de la imagen.

$$xB_i = \{ColorB, t, r_c\} = \left\{ (R - G) \frac{R}{B}, t, e^{\frac{r}{k}} \right\}$$

Los valores del vector de características no se normalizan, esto se hace a conciencia, a fin de dotar de mayor o menor peso a las características dentro de la clasificación.

Para cada uno de estos grupos se van a comparar cinco clasificadores diferentes:

- Clasificador por distancias euclídeas
- Clasificador por número de vecinos, con $k=25$ (10% de la población original total en la base de datos)
- Clasificador híbrido, como producto del número de vecinos por la distancia euclídea normalizada restada a la unidad
- Clasificador híbrido como cociente del número de vecinos entre la distancia euclídea normalizada
- Clasificador bayesiano

El código desarrollado para clasificar se encuentra en el anexo 7.4, incluyendo las cinco funciones de clasificación.

De esta manera se obtendrán un total de 15 tablas de valores, donde se indicará el éxito o fracaso de cada clasificación. Es muy importante lograr un porcentaje de éxito lo más cercano al 100% para lograr un sistema de clasificación que pueda prescindir de supervisión manual. Por esa razón en este proyecto se

ha optado por comprobar un gran número de clasificadores y varios vectores de características diferentes a fin de lograr el menor número de errores.

Las tablas de resultados se pueden ver en el anexo 7.3, donde el color verde significa que ha clasificado correctamente el sujeto, y el color rojo que no ha sido así.

A continuación, se muestran los resultados obtenidos, empleando un total de 50 imágenes de langostinos que se han repartido de la siguiente forma:

- 8 de clase PK
- 6 de clase PM
- 7 de clase PP
- 15 de clase PV
- 14 de clase J

Tabla 3. Porcentaje de acierto según método, vector y tipo de langostino

		PK	PM	PP	PV	J
Vector O	<i>D. euclidea</i>	62.50%	66.67%	71.43%	80.00%	85.71%
	<i>k-NN (k=25)</i>	62.50%	83.33%	85.71%	86.67%	92.86%
	Híbrido 1	62.50%	83.33%	85.71%	86.67%	92.86%
	Híbrido 2	62.50%	83.33%	85.71%	86.67%	92.86%
	Bayesiano	87.50%	0.00%	0.00%	86.67%	100.00%
Vector A	<i>D. euclidea</i>	37.50%	66.67%	28.57%	66.67%	71.43%
	<i>k-NN (k=25)</i>	12.50%	66.67%	71.43%	66.67%	92.86%
	Híbrido 1	12.50%	66.67%	71.43%	66.67%	92.86%
	Híbrido 2	37.50%	66.67%	57.14%	66.67%	85.71%
	Bayesiano	87.50%	83.33%	85.71%	93.33%	100.00%
Vector B	<i>D. euclidea</i>	37.50%	100.00%	57.14%	86.67%	85.71%
	<i>k-NN (k=25)</i>	37.50%	100.00%	57.14%	93.33%	85.71%
	Híbrido 1	50.00%	100.00%	57.14%	100.00%	85.71%
	Híbrido 2	37.50%	100.00%	57.14%	100.00%	85.71%
	Bayesiano	100.00%	100.00%	100.00%	100.00%	100.00%

Lo primero que salta a la vista en base a los resultados es el elevado porcentaje de éxito para prácticamente todos los casos de los langostinos de clase J frente al resto de clases. Esto es debido a su clara diferenciación por su tamaño medio, del orden de tres a cinco veces superior.

El vector "O", basado directamente en las 5 características obtenidas de la imagen, ofrece unos resultados más o menos estables en los cuatro primeros tipos de clasificadores, donde únicamente PK resulta tener una tasa de éxito excesivamente reducida. Por otra parte, en dicho vector "O", el clasificador por probabilidad bayesiana resulta ser un erróneo, con dos de las cinco clases de langostino donde la tasa de acierto es nula.

La clase PK, especialmente empleando el vector de características "A", quedaba poco definido en el espacio de características, tendiendo a confundirse con otros clústeres vecinos como PM, PP y PV. Por

esta razón el vector de características “A” ofrece unos resultados más pobres en general, para todos los clasificadores, salvo quizás el bayesiano, donde los resultados son aceptables dada la naturaleza probabilística de este método, menos arbitraria que la de los otros.

Atendiendo al vector de características “B”, los resultados son muy buenos a excepción de las clases PK y PV, las cuales se superponen en el espacio, y generan una tasa de fallo elevada entre los dos. A priori parece que el vector “B” con un clasificador bayesiano, es la opción idónea, al haber tenido una tasa de fallo nula. Este notable éxito se debe a una correcta y cuidada agrupación de los valores mediante las operaciones matemáticas que se han ido definiendo a lo largo del proyecto, cuyo objetivo ha sido precisamente esto.

Se muestra en la tabla 4 el resumen de porcentaje de acierto para cada clasificador y cada vector de características.

Tabla 4. Porcentaje de acierto según método y vector para el conjunto de langostinos

	Vector O	Vector A	Vector B
<i>D. euclídea</i>	76.00%	58.00%	76.00%
<i>k-NN (k=25)</i>	84.00%	66.00%	78.00%
<i>Híbrido 1</i>	84.00%	66.00%	80.00%
<i>Híbrido 2</i>	84.00%	66.00%	80.00%
<i>Bayesiano</i>	68.00%	92.00%	100.00%

El método basado únicamente en la distancia euclídea no resulta fiable, debido principalmente a su poca o escasa capacidad de adaptación al caso, ya que su decisión es binaria y completamente preestablecida en función de la región espacial donde se encuentre. Si además los clústeres no están suficientemente alejados y sus valores se solapan como ocurre especialmente en el caso del vector A, las clasificaciones resultantes no son válidas.

La clasificación por vecinos cercanos muestra una mejora de resultados en todos los casos frente a la distancia euclídea, especialmente notable en los vectores O y A, siendo en este último la tasa de acierto inferior al 70%. Se podría ajustar el número de vecinos tomados en cuenta para los cálculos, pero un 10% del total inicial, a juicio del autor y las diferentes pruebas empíricas realizadas, parece correcta aproximación.

La diferencia entre los dos híbridos definidos, si bien en los resultados pormenorizados que se encuentran en el anexo correspondiente sí muestran diferencias en determinadas clasificaciones, globalmente son iguales en cuanto a los resultados de la clasificación. En cualquier caso, su tasa de acierto no dista prácticamente de los clasificadores en que se basa, aunque mejora ligeramente en el caso del vector B.

El clasificador bayesiano ofrece resultados pobres cuando se tienen en cuenta las cinco características “en bruto” de la imagen, pero aumenta en gran medida la tasa de éxito cuando estos valores son simplificados y procesados hasta converger en tres. Adicionalmente, en el caso del vector B frente al vector A, se observa una sustancial mejora debido a una agrupación más lograda en el primero. En concreto, se han obtenido muy buenos resultados en la clasificación con este método, con una tasa de fallo nula en el caso del vector que se ha tratado de optimizar y encontrar a lo largo del proyecto, el vector de características B.

Página en blanco intencionadamente

5. CONCLUSIONES Y FUTUROS TRABAJOS

En la actualidad no existe un sistema clasificador automático para las especies de langostinos, aun cuando estos se pescan mezclados en muchos caladeros. La única clasificadora automatizada que existe es un sistema mecánico de rodillos que separa los especímenes por tamaño, una vez han sido discriminados de forma manual por especie.

Se ha propuesto en el proyecto una posible alternativa a este sistema manual y mecánico, a través de la visión artificial o por computación, de manera que el propio sistema sea capaz de evaluar, medir y clasificar la muestra en función de sus características visuales.

El proceso, dividido en tres fases claramente diferenciadas, comienza por una separación de la muestra evaluada, umbralizando y discretizándola del ambiente que la rodea; es decir, de la cinta transportadora o bandeja en que se encuentra. Para ello primero se ha definido una umbralización basada en un valor límite de la escala de grises (calculado de forma dinámica mediante Otsu), y una diferencia de color entre el valor característico en la mayoría de las especies: R y con un valor más reducido B.

El siguiente punto de esta fase ha sido completar la información de la imagen mediante la probabilidad de pertenencia a un conjunto (langostino) u otro (fondo) de cada píxel. Para ello se ha desarrollado un sistema propio basado en probabilidad de pertenencia al langostino, para lo cual se analiza las regiones adyacentes en un área definida para los puntos de la imagen, y se decide, en función del número de estos, la probabilidad de que el punto analizado pertenezca al conjunto fondo o langostino. Así podrá decidirse qué puntos son langostino tras indicar los valores límite a partir de los cuales el programa deberá definir dichos puntos como una cosa u otra.

De esta manera se han logrado unos resultados satisfactorios en todos los casos estudiados, eliminando restos o desperdicios que pueda haber en la imagen de otros especímenes, y completando o corrigiendo determinados puntos que habían quedado fuera. Se termina así la primera fase.

Tras el buen resultado en la discretización de la imagen mediante umbralización y probabilidad de pertenencia, se han extraído los vectores de características de esta. Comienza de esta forma la segunda fase.

Las características elegidas como identificadoras del espécimen han sido el tamaño, el color y el porcentaje de rayado (factor diferenciador de dos grandes grupos de langostinos en la mar).

El tamaño es prácticamente inmediato, y a raíz de éste, además de su utilidad en la clasificación por especie, podrían definirse valores límite a partir de los cuales una posible planta clasificadora los dirigiese en una dirección u otra, basándose no solamente en la especie si no también en las dimensiones.

Para extraer el color se ha tomado el valor medio de las componentes R, G y B a lo largo de toda la muestra del conjunto langostino.

Para obtener el porcentaje de rayado, se ha desarrollado un sistema que se fundamenta en los gradientes direccionales de la imagen, tomando no únicamente los valores inmediatamente adyacentes si no una serie de valores en cada dirección. Este sistema se ha comparado con uno tradicional de umbralización, obteniendo resultados mucho más característicos en el caso del sistema por gradiente.

La última y tercera fase consiste en la clasificación, una vez discretizada la imagen y obtenido el vector de características de esta, que consiste en cinco valores: tamaño, porcentaje de rayado y las tres componentes del color RGB.

En general se recomienda que un vector de características sea capaz de resumir la mayor información en el menor número de ellas [12]. Por ello se ha tratado de buscar vectores alternativos al vector de características bruto o inicial donde se tienen en cuenta los cinco valores mencionados.

Se propone una alternativa única para el rayado, debido a su alta dispersión en el espacio, mediante una función exponencial.

En el caso del color se muestran dos alternativas; la primera, más simple, basada en la relación existente de color y escala de grises de acuerdo al sistema NTSC. La segunda ha sido desarrollada por el autor, basándose en las diferentes relaciones entre componentes R, G y B de la base de datos, a fin de lograr una mayor segregación y diferenciación entre clústeres, sin falsear ni perder información.

Finalmente, terminado el proceso de extracción de características y obtenidos los tres vectores para estudio, se han ensayado cinco clasificadores diferentes, a fin de encontrar aquel donde la tasa de éxito sea más elevada.

Las conclusiones de los resultados se pueden ver en el capítulo 4, siendo el vector buscado y definido "B" junto con el clasificador bayesiano los que han dado una tasa de éxito mejor para las 50 muestras analizadas.

Sin embargo, gracias a los resultados obtenidos con este vector en el resto de los clasificadores, se podría extraer como conclusión que la diferenciación entre las clases PK y PV es claramente mejorable, lo que podría terminar convergiendo en una clasificación no exitosa para el espécimen. Dado que la primera clase pertenece a un grupo de langostinos con rayado superficial, y PV no, sería ésta la aproximación que debería corregirse, no tanto sobre el vector de características como sobre la propia obtención del porcentaje de rayado sobre la imagen. Según puede observarse en las tablas de características de la base de datos [anexo 7.1], el porcentaje de rayado de PV oscila en gran medida. El problema es que la superficie de este tipo de langostino, si bien no tiene rayado, presenta en muchos casos punteado, lo que confunde al método de gradientes definido en el proyecto, por lo que sería éste un posible punto de mejora del resultado.

Queda asimismo probada la efectividad del método de clasificación de langostinos mediante visión artificial, basándose en un vector de características propio, que relaciona las componentes del color a fin de lograr una distribución espacial suficientemente buena para su procesado.

6. BIBLIOGRAFÍA

- [1] <http://www.pescanova.com/ES/content/Ciclo-cria-Langostino>
- [2] <http://www.fao.org>
- [3] <http://www.marypescanoticiaspatagonicas.com/2012/06/encuentran-langostino-de-baja-talla-en-la-subarea-3/>
- [4] <http://maquinasbalzo.com/en/index.php>
- [5] “*Visión por Computador. Fundamentos y métodos*”. Arturo de la Escalera. Prentice Hall, 2001
- [6] <http://www.lemkevision.com/en/>
- [7] <https://www.ojodigital.com/foro/forum.php>
- [8] www.wikipedia.org
- [9] Jorge Valverde Rebaza, “Detección de bordes mediante el algoritmo de Canny”. Universidad Nacional de Trujillo, 2007
- [10] Nobuyuki Otsu, “A threshold selection method from gray-level histogram”, IEEE Transactions on System Man Cybernetics, Vol. SMC-9, No. 1, 1979.
- [11] Luis Enrique Sucar, “Clasificadores Bayesianos: de datos a conceptos”, Instituto Nacional de Astrofísica, Óptica y Electrónica, Tonantzintla, Puebla, México, 2013.
- [12] Eduardo Fernández Cantalejo Padial, “Diseño de un sistema automático de identificación de nubes basado en visión artificial”, Escuela Superior de Ingenieros de Sevilla, Sevilla, España, 2016.
- [13] Eduardo Morales, “Aprendizaje bayesiano”, Instituto Nacional de Astrofísica, Óptica y Electrónica, Tonantzintla, Puebla, México.
- [14] Alvaro Romero, Alejandro Marín, Jovani Alberto Jiménez, “Sistema de clasificación por visión artificial de mangos tipo Tommy”, Facultad de Minas, Universidad Nacional de Colombia, 2015.
- [15] <https://www.itu.int>
- [16] Gonzalo, P. M., & de la Cruz Jesus, M., “Vision por computador. Imagenes digitales y aplicaciones”. Editorial AlfaOmega, 2008
- [17] Macedo-Cruz, A., Pajares-Martinsanz, G., & Santos-Peñas, M. (2010). Clasificación no supervisada con imágenes a color de cobertura terrestre. *Agrociencia*, 44(6), 711-722.
- [18] Farias, G., Dormido-Canto, S., Vega, J., Sánchez, J., Duro, N., Dormido, R., ... & Pajares, G. (2006). Searching for patterns in TJ-II time evolution signals. *Fusion engineering and design*, 81(15-17), 1993-1997.
- [19] Ayed, I., Moyà-Alcover, B., Martínez-Bueso, P., Varona, J., Ghazel, A., & Jaume-i-Capó, A. (2017). Validación de dispositivos RGBD para medir terapéuticamente el equilibrio: el test de alcance funcional con Microsoft Kinect. *Revista Iberoamericana de Automática e Informática Industrial RIAI*, 14(1), 115-120.

- [20] Macedo-Cruz, A., Pajares, G., Santos, M., & Villegas-Romero, I. (2011). Digital image sensor-based assessment of the status of oat (*Avena sativa* L.) crops after frost damage. *Sensors*, *11*(6), 6015-6036.
- [21] Vega, J., Rattá, G., Murari, A., Castro, P., Dormido-Canto, S., Dormido, R., ... & Pastor, I. (2007, October). Recent results on structural pattern recognition for Fusion massive databases. In *2007 IEEE International Symposium on Intelligent Signal Processing* (pp. 1-6). IEEE.
- [22] Santos, M., & Cantos, A. J. (2010). Classification of plasma signals by genetic algorithms. *Fusion Science and Technology*, *58*(2), 706-713.
- [23] Canny, Jon F., "Finding edges and lines in images". 1983
- [24] Piorno Campo, Juan, "Diseño de un nuevo clasificador supervisado para minería de datos", Facultad de informática, Universidad Complutense de Madrid, España, 2009
- [25] D.J. White, C. Svelling, N.J.C. Strachan, "Automated measurement of species and length of fish by computer vision", University of Aberdeen, United Kingdom, 2006
- [26] Mingming Hao, Helong Yu, Daoliang Li, "The Measurement of Fish Size by Machine Vision - A Review", College of Information Technology, Jilin Agricultural University, Changchun, China, 2016
- [27] Joao C. Ferreira, Jorge Branquinho, Paulo Chaves Ferreira, Fernando Piedade, "Computer Vision Algorithms Fishing Vessel Monitoring—Identification of Vessel Plate Number", Instituto Universitário de Lisboa, Portugal, 2017

7. ANEXOS

7.1. BASE DE DATOS

n	PK							
	Rayado	eR	T	R	G	B	Gris	(R-G)*R/B
1	10.55	1.53	2.60	94	54	35	63.78	107.43
2	15.93	1.89	2.49	86	40	33	52.95	119.88
3	17.76	2.03	2.69	83	38	35	51.10	106.71
4	19.73	2.20	2.61	85	41	37	53.69	101.08
5	16.13	1.91	2.60	94	54	25	62.64	150.40
6	11.05	1.56	2.78	95	54	30	63.51	129.83
7	17.76	2.03	2.46	80	42	29	51.87	104.83
8	13.47	1.71	2.49	86	38	31	51.55	133.16
9	11.54	1.59	2.68	80	36	30	48.46	117.33
10	11.44	1.58	2.61	93	55	27	63.16	130.89
11	12.97	1.68	2.53	92	49	28	59.45	141.29
12	11.69	1.60	2.56	93	49	27	59.64	151.56
13	10.51	1.52	2.57	89	43	27	54.92	151.63
14	14.45	1.78	2.63	83	36	29	49.25	134.52
15	13.71	1.73	2.78	83	40	31	51.82	115.13
16	14.06	1.75	2.63	95	52	30	62.34	136.17
17	18.45	2.09	2.52	85	46	36	56.51	92.08
18	14.65	1.80	2.71	89	46	33	57.37	115.97
19	17.56	2.02	2.78	80	34	32	47.52	115.00
20	11.25	1.57	2.57	83	41	34	52.75	102.53
21	19.39	2.17	2.46	92	51	28	60.63	134.71
22	11.29	1.57	2.76	83	37	35	50.52	109.09
23	11.84	1.61	2.52	80	32	30	46.12	128.00
24	15.44	1.85	2.51	81	40	26	50.65	127.73
25	18.01	2.05	2.66	88	49	37	59.28	92.76
26	17.46	2.01	2.74	86	46	32	56.36	107.50
27	19.39	2.17	2.46	90	45	27	56.39	150.00
28	11.59	1.59	2.51	86	42	36	54.46	105.11
29	18.80	2.12	2.47	87	41	26	53.04	153.92
30	15.05	1.83	2.65	80	32	36	46.80	106.67
31	15.64	1.87	2.56	83	39	34	51.58	107.41
32	15.79	1.88	2.80	92	53	33	62.37	108.73
33	14.90	1.81	2.74	88	45	31	56.25	122.06
34	10.26	1.51	2.53	87	46	28	56.20	127.39
35	12.04	1.62	2.70	89	48	26	57.74	140.35
36	14.95	1.82	2.61	92	49	24	59.00	164.83
37	15.14	1.83	2.79	91	50	37	60.77	100.84
38	17.32	2.00	2.53	95	52	37	63.14	110.41
39	13.96	1.75	2.47	82	35	28	48.25	137.64
40	19.78	2.21	2.62	80	32	29	46.00	132.41
41	18.70	2.11	2.76	95	55	27	63.76	140.74
42	18.80	2.12	2.66	95	48	24	59.31	186.04
43	12.63	1.66	2.78	94	46	29	58.40	155.59
44	15.39	1.85	2.67	89	50	26	58.92	133.50
45	19.78	2.21	2.47	80	37	31	49.17	110.97
46	13.17	1.69	2.75	90	52	26	60.39	131.54
47	20.08	2.23	2.77	94	50	25	60.30	165.44
48	15.69	1.87	2.47	81	39	30	50.52	113.40
49	13.52	1.72	2.80	87	47	36	57.70	96.67
50	18.50	2.10	2.67	93	49	30	59.98	136.40

n	PM							
	Rayado	eR	T	R	G	B	Gris	(R-G)*R/B
1	6.12	1.28	1.88	77	45	27	52.51	91.26
2	9.35	1.45	2.53	74	43	34	51.24	67.47
3	4.27	1.19	2.44	72	44	30	50.77	67.20
4	5.21	1.23	1.94	70	38	32	46.88	70.00
5	7.65	1.36	2.28	78	47	32	54.55	75.56
6	9.24	1.45	2.12	72	45	31	51.47	62.71
7	8.26	1.39	1.84	71	44	33	50.81	58.09
8	6.27	1.28	2.23	75	45	33	52.59	68.18
9	7.25	1.34	2.10	73	43	30	50.48	73.00
10	7.86	1.37	1.92	77	47	29	53.91	79.66
11	6.48	1.30	2.34	75	42	28	50.26	88.39
12	5.00	1.22	2.43	73	41	27	48.96	86.52
13	7.50	1.35	1.99	65	38	25	44.58	70.20
14	3.80	1.16	2.37	70	42	26	48.54	75.38
15	7.10	1.33	2.54	78	48	25	54.34	93.60
16	8.63	1.41	2.14	70	38	32	46.88	70.00
17	4.02	1.17	2.55	77	48	30	54.61	74.43
18	3.36	1.14	2.49	69	39	29	46.82	71.38
19	7.25	1.34	2.01	73	45	31	51.77	65.94
20	5.29	1.24	2.42	74	42	27	49.85	87.70
21	9.17	1.44	1.82	69	41	25	47.54	77.28
22	3.18	1.14	2.37	66	34	34	43.56	62.12
23	5.94	1.27	2.07	67	40	30	46.93	60.30
24	9.42	1.46	2.22	73	46	31	52.36	63.58
25	6.77	1.31	2.38	73	44	27	50.73	78.41
26	4.31	1.19	2.26	68	37	31	45.58	68.00
27	8.26	1.39	1.81	71	43	33	50.22	60.24
28	5.94	1.27	2.20	71	42	28	49.07	73.54
29	9.46	1.46	2.51	76	44	29	51.85	83.86
30	7.57	1.35	1.78	69	42	32	48.93	58.22
31	6.19	1.28	2.51	78	45	26	52.69	99.00
32	8.44	1.40	2.31	71	41	30	48.71	71.00
33	6.19	1.28	2.41	72	39	33	48.18	72.00
34	6.74	1.31	2.41	69	41	34	48.57	56.82
35	9.42	1.46	1.82	74	43	33	51.12	69.52
36	10.04	1.49	2.55	69	36	34	45.63	66.97
37	7.10	1.33	1.81	65	37	31	44.68	58.71
38	9.42	1.46	1.84	73	46	25	51.67	78.84
39	7.86	1.37	2.11	73	44	32	51.30	66.16
40	8.81	1.42	2.19	71	43	27	49.54	73.63
41	4.34	1.19	2.12	65	37	28	44.34	65.00
42	9.82	1.48	1.90	76	47	31	53.84	71.10
43	9.90	1.49	2.19	76	48	27	53.97	78.81
44	4.52	1.20	2.02	69	41	31	48.23	62.32
45	7.72	1.36	2.52	71	43	32	50.11	62.13
46	6.92	1.32	2.05	73	42	25	49.32	90.52
47	3.69	1.16	2.37	78	49	32	55.73	70.69
48	9.24	1.45	2.21	76	46	32	53.37	71.25
49	8.59	1.41	2.13	78	45	34	53.61	75.71
50	7.83	1.37	1.80	73	42	26	49.44	87.04

n	PP							
	Rayado	eR	T	R	G	B	Gris	(R-G)*R/B
1	10.40	1.52	3.59	96	53	34	63.68	121.41
2	10.47	1.52	3.73	90	45	25	56.17	162.00
3	7.27	1.34	2.95	91	45	27	56.69	155.04
4	6.11	1.28	3.57	93	52	32	61.97	119.16
5	9.49	1.46	3.55	96	56	29	64.87	132.41
6	6.72	1.31	3.63	97	50	31	61.88	147.06
7	10.06	1.50	3.06	97	53	33	63.87	129.33
8	6.93	1.32	2.78	96	53	26	62.77	158.77
9	9.20	1.44	3.31	88	41	33	54.13	125.33
10	6.06	1.27	3.08	89	50	31	59.49	111.97
11	8.81	1.42	3.12	96	51	28	61.82	154.29
12	9.20	1.44	3.19	96	51	26	61.60	166.15
13	7.73	1.36	3.53	87	47	27	56.67	128.89
14	8.11	1.38	3.46	97	52	28	62.71	155.89
15	10.62	1.53	2.96	97	58	28	66.23	135.11
16	6.69	1.31	3.11	96	57	23	64.78	162.78
17	6.35	1.29	2.90	96	49	24	60.19	188.00
18	10.40	1.52	2.82	88	45	34	56.59	111.29
19	6.11	1.28	3.66	94	54	25	62.64	150.40
20	9.25	1.45	3.26	93	51	32	61.38	122.06
21	8.59	1.41	3.60	95	51	26	61.30	160.77
22	8.33	1.40	2.95	95	51	33	62.09	126.67
23	9.61	1.47	3.31	89	47	30	57.61	124.60
24	8.21	1.39	3.35	90	42	25	54.41	172.80
25	6.86	1.32	3.36	95	46	29	58.70	160.52
26	7.17	1.33	3.61	91	45	27	56.69	155.04
27	10.57	1.53	2.82	91	42	32	55.50	139.34
28	9.97	1.49	3.10	92	45	32	57.56	135.13
29	7.73	1.36	2.87	95	49	26	60.12	168.08
30	8.93	1.43	3.46	96	57	29	65.46	129.10
31	8.38	1.40	2.96	91	42	33	55.62	135.12
32	10.11	1.50	2.84	92	53	28	61.80	128.14
33	10.47	1.52	3.64	93	49	31	60.09	132.00
34	8.45	1.40	3.65	88	45	34	56.59	111.29
35	9.08	1.44	2.87	89	42	33	55.02	126.76
36	9.37	1.45	3.39	96	49	26	60.42	173.54
37	6.16	1.28	2.90	87	38	24	51.05	177.63
38	6.23	1.28	3.50	90	45	23	55.94	176.09
39	9.99	1.49	2.95	88	45	29	56.02	130.48
40	9.82	1.48	2.91	92	52	27	61.10	136.30
41	8.43	1.40	3.71	94	50	31	60.98	133.42
42	8.35	1.40	3.45	93	48	23	58.60	181.96
43	9.34	1.45	3.80	89	47	30	57.61	124.60
44	8.79	1.42	3.70	94	46	25	57.95	180.48
45	7.17	1.33	3.54	96	49	30	60.88	150.40
46	6.84	1.31	3.46	91	46	26	57.17	157.50
47	9.78	1.48	3.31	92	53	24	61.35	149.50
48	8.86	1.43	2.95	90	51	23	59.46	152.61
49	7.37	1.34	3.25	90	49	30	59.08	123.00
50	9.22	1.45	3.37	92	49	32	59.91	123.63

n	PV							
	Rayado	eR	T	R	G	B	Gris	(R-G)*R/B
1	6.53	1.30	1.85	73	52	54	58.50	28.39
2	8.51	1.41	2.12	87	67	41	70.01	42.44
3	8.10	1.38	1.07	84	62	49	67.09	37.71
4	8.30	1.39	1.23	78	56	45	61.32	38.13
5	3.95	1.17	2.01	90	66	49	71.23	44.08
6	11.16	1.56	1.56	84	60	56	66.71	36.00
7	10.62	1.53	1.83	97	73	47	77.20	49.53
8	8.24	1.39	1.41	83	58	42	63.64	49.40
9	3.95	1.17	1.88	76	57	49	61.76	29.47
10	4.29	1.19	1.69	81	59	46	64.09	38.74
11	13.27	1.70	1.71	74	49	44	55.90	42.05
12	11.37	1.58	2.00	97	76	59	80.33	34.53
13	7.08	1.33	1.32	73	49	51	56.40	34.35
14	4.01	1.17	2.14	83	59	43	64.34	46.33
15	4.76	1.21	1.58	81	61	47	65.38	34.47
16	12.73	1.66	2.04	90	65	46	70.30	48.91
17	7.21	1.33	1.89	83	61	55	66.89	33.20
18	1.90	1.08	1.29	81	57	60	64.51	32.40
19	11.91	1.61	1.79	81	63	57	67.69	25.58
20	5.03	1.22	1.84	92	72	48	75.23	38.33
21	1.83	1.08	1.62	86	68	59	72.35	26.24
22	9.94	1.49	1.89	95	75	62	79.49	30.65
23	0.95	1.04	1.20	80	55	59	62.92	33.90
24	0.47	1.02	1.32	94	70	63	76.37	35.81
25	7.01	1.32	1.15	95	72	54	76.82	40.46
26	5.51	1.25	1.28	75	52	44	57.96	39.20
27	7.69	1.36	1.72	79	59	50	63.95	31.60
28	4.22	1.18	1.23	86	64	43	68.18	44.00
29	3.06	1.13	2.04	82	59	53	65.18	35.58
30	5.78	1.26	1.83	86	65	51	69.67	35.41
31	10.21	1.50	1.48	98	79	60	82.51	31.03
32	2.45	1.10	1.08	93	68	56	74.10	41.52
33	3.20	1.14	2.06	88	69	60	73.65	27.87
34	2.24	1.09	1.72	78	53	48	59.90	40.63
35	6.40	1.29	1.84	88	67	60	72.47	30.80
36	10.55	1.53	2.01	86	65	47	69.22	38.43
37	9.33	1.45	1.40	90	71	56	74.96	30.54
38	5.72	1.26	2.13	82	57	50	63.67	41.00
39	7.01	1.32	1.31	85	61	42	66.00	48.57
40	2.45	1.10	2.04	86	67	60	71.87	27.23
41	5.65	1.25	2.09	87	65	43	69.06	44.51
42	13.41	1.71	1.55	80	62	50	66.01	28.80
43	6.12	1.28	1.09	91	69	59	74.43	33.93
44	13.68	1.73	1.54	87	62	41	67.07	53.05
45	12.12	1.62	1.68	73	49	44	55.60	39.82
46	1.29	1.05	1.59	96	73	59	78.27	37.42
47	6.81	1.31	1.55	78	53	45	59.56	43.33
48	5.38	1.24	1.76	72	47	63	56.29	28.57
49	6.67	1.31	1.90	87	63	43	67.89	48.56
50	8.58	1.41	2.06	78	53	62	61.49	31.45

J								
n	Rayado	eR	T	R	G	B	Gris	(R-G)*R/B
1	21.47	2.36	12.06	44	34	42	37.90	10.48
2	18.76	2.12	8.72	55	46	37	47.66	13.38
3	18.44	2.09	10.83	44	34	41	37.78	10.73
4	20.46	2.27	8.63	47	37	36	39.87	13.06
5	14.36	1.78	10.79	44	32	25	34.79	21.12
6	21.33	2.35	10.46	46	34	27	36.79	20.44
7	21.74	2.39	10.06	56	47	25	47.18	20.16
8	13.95	1.75	12.35	58	42	25	44.84	37.12
9	17.02	1.98	12.24	40	27	28	31.00	18.57
10	18.44	2.09	10.57	46	30	31	34.89	23.74
11	15.96	1.89	9.40	43	33	34	36.10	12.65
12	15.74	1.88	9.75	51	38	42	42.34	15.79
13	21.79	2.39	8.92	49	36	34	39.65	18.74
14	14.31	1.77	9.51	60	51	27	50.95	20.00
15	15.51	1.86	10.98	43	29	35	33.87	17.20
16	19.86	2.21	8.21	50	41	24	41.75	18.75
17	21.74	2.39	10.50	58	44	26	46.13	31.23
18	15.09	1.83	11.27	51	39	20	40.42	30.60
19	22.20	2.43	11.07	43	30	36	34.57	15.53
20	21.51	2.36	8.54	40	26	34	31.09	16.47
21	20.92	2.31	12.17	43	34	22	35.32	17.59
22	13.49	1.72	9.60	39	26	43	31.82	11.79
23	14.73	1.80	12.11	50	41	23	41.63	19.57
24	17.11	1.98	11.05	42	31	40	35.31	11.55
25	13.72	1.73	10.21	56	45	27	46.23	22.81
26	16.65	1.95	12.11	57	48	22	47.72	23.32
27	16.70	1.95	12.13	39	29	29	31.99	13.45
28	15.60	1.87	8.96	56	46	40	48.30	14.00
29	18.12	2.06	11.01	58	45	39	48.20	19.33
30	20.82	2.30	10.70	58	46	43	49.24	16.19
31	18.49	2.09	12.22	54	44	23	44.59	23.48
32	14.22	1.77	9.27	40	25	26	29.60	23.08
33	16.24	1.91	11.89	52	40	27	42.10	23.11
34	13.35	1.71	8.94	48	34	39	38.75	17.23
35	21.01	2.32	11.64	37	27	20	29.19	18.50
36	14.68	1.80	10.08	60	49	31	50.23	21.29
37	20.18	2.24	9.93	57	43	26	45.24	30.69
38	18.53	2.10	9.00	50	37	28	39.86	23.21
39	15.69	1.87	10.92	49	38	28	40.14	19.25
40	20.37	2.26	11.64	49	34	33	38.37	22.27
41	13.72	1.73	11.78	48	37	41	40.74	12.88
42	19.45	2.18	8.30	38	28	24	30.53	15.83
43	15.37	1.85	11.82	39	28	34	31.97	12.62
44	14.50	1.79	9.00	43	34	43	37.71	9.00
45	20.14	2.24	11.62	54	41	43	45.11	16.33
46	15.92	1.89	11.25	54	40	37	43.84	20.43
47	17.80	2.04	11.78	60	46	22	47.44	38.18
48	13.72	1.73	8.39	58	42	37	46.21	25.08
49	16.88	1.96	10.87	39	23	28	28.35	22.29
50	13.86	1.74	10.10	37	28	32	31.14	10.41

7.2. IMÁGENES PROCESADAS

Imágenes procesadas									
n	Tipo	Ray	eR	Tamaño	R	G	B	RGRB	gris
1	PP	8.34	1.40	3.14	96	53	32	129.0	63.45
2	PV	8.46	1.40	1.41	87	66	53	34.5	70.79
3	PV	12.75	1.67	1.69	77	53	41	45.1	58.80
4	PK	10.99	1.55	2.46	85	43	29	123.1	53.95
5	PP	6.18	1.28	3.71	90	42	24	180.0	54.29
6	J	18.05	2.06	9.40	58	45	41	18.4	48.43
7	J	17.65	2.03	9.46	56	40	38	23.6	44.55
8	PM	9.86	1.48	1.96	80	48	34	75.3	55.96
9	J	16.69	1.95	12.19	44	32	28	18.9	35.13
10	PV	6.48	1.30	1.59	99	77	64	34.0	82.09
11	PV	4.31	1.19	1.60	97	65	51	60.9	72.96
12	J	22.58	2.47	8.61	52	33	31	31.9	38.45
13	PK	18.29	2.08	2.56	92	49	35	113.0	60.25
14	PV	3.34	1.14	1.36	98	71	58	45.6	77.58
15	PK	10.21	1.50	2.62	93	46	32	136.6	58.45
16	J	23.46	2.56	8.53	58	45	41	18.4	48.43
17	PM	2.60	1.11	1.82	67	37	26	77.3	44.71
18	PP	10.80	1.54	3.03	87	48	31	109.5	57.71
19	PV	17.18	1.99	1.88	80	64	54	23.7	67.64
20	PP	8.56	1.41	2.65	124	68	50	138.88	82.68
21	J	19.18	2.15	10.89	60	46	44	19.1	49.95
22	PM	9.18	1.44	2.20	77	44	32	79.4	52.49
23	J	15.19	1.84	10.04	42	31	27	17.1	33.83
24	PK	14.49	1.79	2.81	77	35	23	140.6	46.18
25	PV	6.48	1.30	1.43	71	54	44	27.4	57.94
26	J	16.28	1.92	12.93	37	28	26	12.8	30.46
27	PK	15.48	1.86	2.75	88	45	31	122.1	56.25
28	PV	7.81	1.37	1.49	78	61	51	26.0	64.94
29	J	17.61	2.02	9.38	74	60	56	18.50	63.72
30	PK	12.34	1.64	2.68	124	90	83	50.80	99.36
31	J	18.16	2.07	11.94	57	37	32	35.6	42.40
32	J	16.89	1.97	9.10	36	22	20	25.2	25.95
33	PP	8.06	1.38	3.21	91	46	28	146.3	57.39
34	PK	20.08	2.23	2.58	91	44	29	147.5	56.33
35	PM	8.07	1.38	2.05	66	36	24	82.5	43.60
36	PP	5.98	1.27	2.80	97	45	22	229.3	57.92
37	J	14.29	1.77	10.07	41	31	28	14.6	33.64
38	PV	5.72	1.26	1.43	81	63	51	28.6	67.01
39	PV	5.54	1.25	2.11	94	68	57	42.9	74.51
40	PV	8.00	1.38	1.48	85	66	54	29.9	70.30
41	J	18.53	2.10	11.12	46	34	32	17.3	37.36
42	PV	5.86	1.26	1.84	76	55	44	36.3	60.02
43	PV	8.85	1.42	2.10	65	49	36	28.89	52.30
44	PV	14.97	1.82	1.32	65	46	38	32.50	50.76
45	PM	6.19	1.28	1.78	86	53	41	69.22	61.49
46	PP	10.04	1.49	3.83	91	52	34	104.4	61.60
47	PK	16.47	1.93	2.64	85	49	37	82.7	58.39
48	J	16.22	1.91	9.37	44	33	29	16.7	35.83
49	PM	3.80	1.16	2.66	69	42	31	60.1	48.81
50	PV	6.60	1.30	1.68	92	70	56	36.1	74.97

7.3. RESULTADOS

7.3.1. VECTOR $xO_i = \{Ray, t, R, G, B\}$

Distancias euclídeas; $xO_i = \{Ray, t, R, G, B\}$						
n	Tipo	PK	PM	PP	PV	J
1	PP	0.114	0.213	0.053	0.200	0.421
2	PV	0.202	0.228	0.194	0.032	0.344
3	PV	0.161	0.157	0.194	0.158	0.329
4	PK	0.055	0.138	0.102	0.309	0.396
5	PP	0.099	0.158	0.075	0.293	0.375
6	J	0.222	0.155	0.266	0.249	0.108
7	J	0.227	0.152	0.273	0.277	0.070
8	PM	0.113	0.114	0.148	0.249	0.377
9	J	0.234	0.168	0.267	0.291	0.039
10	PV	0.197	0.223	0.185	0.097	0.297
11	PV	0.186	0.226	0.163	0.074	0.351
12	J	0.225	0.164	0.269	0.299	0.044
13	PK	0.073	0.209	0.102	0.219	0.398
14	PV	0.195	0.225	0.178	0.083	0.319
15	PK	0.072	0.199	0.045	0.252	0.434
16	J	0.215	0.165	0.262	0.247	0.110
17	PM	0.197	0.075	0.222	0.309	0.198
18	PP	0.060	0.170	0.069	0.268	0.434
19	PV	0.191	0.216	0.166	0.061	0.366
20	PP	0.174	0.221	0.155	0.145	0.306
21	J	0.217	0.163	0.260	0.230	0.130
22	PM	0.130	0.061	0.177	0.296	0.335
23	J	0.233	0.167	0.263	0.287	0.049
24	PK	0.130	0.110	0.181	0.331	0.249
25	PV	0.205	0.149	0.223	0.144	0.279
26	J	0.229	0.173	0.256	0.276	0.067
27	PK	0.009	0.184	0.097	0.290	0.419
28	PV	0.209	0.206	0.213	0.049	0.323
29	J	0.205	0.207	0.224	0.108	0.257
30	PK	0.194	0.217	0.186	0.143	0.261
31	J	0.222	0.149	0.268	0.303	0.058
32	J	0.221	0.171	0.246	0.275	0.087
33	PP	0.080	0.182	0.031	0.279	0.429
34	PK	0.052	0.193	0.107	0.276	0.372
35	PM	0.192	0.084	0.228	0.323	0.174
36	PP	0.113	0.187	0.065	0.262	0.373
37	J	0.234	0.167	0.263	0.284	0.051
38	PV	0.211	0.216	0.206	0.027	0.340
39	PV	0.197	0.227	0.182	0.065	0.329
40	PV	0.207	0.227	0.200	0.030	0.337
41	J	0.239	0.171	0.276	0.293	0.022
42	PV	0.194	0.164	0.203	0.116	0.325
43	PV	0.209	0.099	0.244	0.242	0.206
44	PV	0.199	0.116	0.251	0.250	0.184
45	PM	0.149	0.186	0.136	0.128	0.401
46	PP	0.098	0.202	0.064	0.203	0.433
47	PK	0.083	0.181	0.136	0.215	0.387
48	J	0.237	0.167	0.270	0.292	0.035
49	PM	0.197	0.044	0.226	0.296	0.237
50	PV	0.199	0.227	0.185	0.062	0.327

k-NN=25; $xO_i = \{Ray, t, R, G, B\}$						
n	Tipo	PK	PM	PP	PV	J
1	PP	4	0	21	0	0
2	PV	0	0	0	25	0
3	PV	2	7	0	16	0
4	PK	12	0	13	0	0
5	PP	3	0	22	0	0
6	J	0	0	0	0	25
7	J	0	0	0	0	25
8	PM	4	19	2	0	0
9	J	0	0	0	0	25
10	PV	0	0	0	25	0
11	PV	0	0	0	25	0
12	J	0	0	0	0	25
13	PK	19	0	6	0	0
14	PV	0	0	0	25	0
15	PK	5	0	20	0	0
16	J	0	0	0	0	25
17	PM	0	25	0	0	0
18	PP	10	0	15	0	0
19	PV	0	0	0	25	0
20	PP	2	0	7	16	0
21	J	0	0	0	2	23
22	PM	0	25	0	0	0
23	J	0	0	0	0	25
24	PK	13	12	0	0	0
25	PV	0	8	0	17	0
26	J	0	0	0	0	25
27	PK	18	0	7	0	0
28	PV	0	0	0	25	0
29	J	0	0	0	25	0
30	PK	0	0	0	25	0
31	J	0	0	0	0	25
32	J	0	0	0	0	25
33	PP	5	0	20	0	0
34	PK	21	0	4	0	0
35	PM	0	25	0	0	0
36	PP	3	0	22	0	0
37	J	0	0	0	0	25
38	PV	0	0	0	25	0
39	PV	0	0	0	25	0
40	PV	0	0	0	25	0
41	J	0	0	0	0	25
42	PV	0	3	0	22	0
43	PV	0	22	0	1	2
44	PV	0	17	0	2	6
45	PM	3	0	7	15	0
46	PP	7	0	18	0	0
47	PK	18	0	7	0	0
48	J	0	0	0	0	25
49	PM	0	25	0	0	0
50	PV	0	0	0	25	0

Híbrido kNN=25; xO _i ={Ray,t,R,G,B}						
n	Tipo	PK	PM	PP	PV	J
1	PP	3.546	0.000	19.887	0.000	0.000
2	PV	0.000	0.000	0.000	24.209	0.000
3	PV	1.677	5.901	0.000	13.472	0.000
4	PK	11.339	0.000	11.675	0.000	0.000
5	PP	2.704	0.000	20.357	0.000	0.000
6	J	0.000	0.000	0.000	0.000	22.303
7	J	0.000	0.000	0.000	0.000	23.248
8	PM	3.550	16.844	1.704	0.000	0.000
9	J	0.000	0.000	0.000	0.000	24.018
10	PV	0.000	0.000	0.000	22.565	0.000
11	PV	0.000	0.000	0.000	23.145	0.000
12	J	0.000	0.000	0.000	0.000	23.910
13	PK	17.619	0.000	5.387	0.000	0.000
14	PV	0.000	0.000	0.000	22.923	0.000
15	PK	4.643	0.000	19.106	0.000	0.000
16	J	0.000	0.000	0.000	0.000	22.253
17	PM	0.000	23.128	0.000	0.000	0.000
18	PP	9.403	0.000	13.964	0.000	0.000
19	PV	0.000	0.000	0.000	23.470	0.000
20	PP	1.652	0.000	5.919	13.686	0.000
21	J	0.000	0.000	0.000	1.539	20.008
22	PM	0.000	23.480	0.000	0.000	0.000
23	J	0.000	0.000	0.000	0.000	23.770
24	PK	11.311	10.686	0.000	0.000	0.000
25	PV	0.000	6.806	0.000	14.549	0.000
26	J	0.000	0.000	0.000	0.000	23.333
27	PK	17.831	0.000	6.321	0.000	0.000
28	PV	0.000	0.000	0.000	23.773	0.000
29	J	0.000	0.000	0.000	22.300	0.000
30	PK	0.000	0.000	0.000	21.435	0.000
31	J	0.000	0.000	0.000	0.000	23.563
32	J	0.000	0.000	0.000	0.000	22.823
33	PP	4.599	0.000	19.388	0.000	0.000
34	PK	19.900	0.000	3.572	0.000	0.000
35	PM	0.000	22.913	0.000	0.000	0.000
36	PP	2.662	0.000	20.561	0.000	0.000
37	J	0.000	0.000	0.000	0.000	23.725
38	PV	0.000	0.000	0.000	24.325	0.000
39	PV	0.000	0.000	0.000	23.365	0.000
40	PV	0.000	0.000	0.000	24.263	0.000
41	J	0.000	0.000	0.000	0.000	24.448
42	PV	0.000	2.509	0.000	19.455	0.000
43	PV	0.000	19.820	0.000	0.758	1.589
44	PV	0.000	15.023	0.000	1.500	4.898
45	PM	2.554	0.000	6.047	13.080	0.000
46	PP	6.315	0.000	16.857	0.000	0.000
47	PK	16.515	0.000	6.049	0.000	0.000
48	J	0.000	0.000	0.000	0.000	24.128
49	PM	0.000	23.908	0.000	0.000	0.000
50	PV	0.000	0.000	0.000	23.453	0.000

Híbrido alternativo kNN=25; xO _i ={Ray,t,R,G,B}						
n	Tipo	PK	PM	PP	PV	J
1	PP	3.522	.	39.610	0.000	0.000
2	PV	0.000	0.000	0.000	78.996	0.000
3	PV	1.239	4.458	0.000	10.126	0.000
4	PK	21.779	0.000	12.758	0.000	0.000
5	PP	3.040	0.000	29.451	0.000	0.000
6	J	0.000	0.000	0.000	0.000	23.170
7	J	0.000	0.000	0.000	0.000	35.663
8	PM	3.556	16.740	1.350	0.000	0.000
9	J	0.000	0.000	0.000	0.000	63.613
10	PV	0.000	0.000	0.000	25.667	0.000
11	PV	0.000	0.000	0.000	33.693	0.000
12	J	0.000	0.000	0.000	0.000	57.339
13	PK	26.135	0.000	5.877	0.000	0.000
14	PV	0.000	0.000	0.000	30.084	0.000
15	PK	6.993	0.000	44.743	0.000	0.000
16	J	0.000	0.000	0.000	0.000	22.748
17	PM	0.000	33.378	0.000	0.000	0.000
18	PP	16.750	0.000	21.708	0.000	0.000
19	PV	0.000	0.000	0.000	40.850	0.000
20	PP	1.150	0.000	4.531	11.065	0.000
21	J	0.000	0.000	0.000	0.868	17.679
22	PM	0.000	41.118	0.000	0.000	0.000
23	J	0.000	0.000	0.000	0.000	50.813
24	PK	10.008	10.959	0.000	0.000	0.000
25	PV	0.000	5.362	0.000	11.789	0.000
26	J	0.000	0.000	0.000	0.000	37.481
27	PK	191.489	0.000	7.216	0.000	0.000
28	PV	0.000	0.000	0.000	50.916	0.000
29	J	0.000	0.000	0.000	23.148	0.000
30	PK	0.000	0.000	0.000	17.532	0.000
31	J	0.000	0.000	0.000	0.000	43.478
32	J	0.000	0.000	0.000	0.000	28.703
33	PP	6.234	0.000	65.359	0.000	0.000
34	PK	40.076	0.000	3.735	0.000	0.000
35	PM	0.000	29.940	0.000	0.000	0.000
36	PP	2.662	0.000	33.639	0.000	0.000
37	J	0.000	0.000	0.000	0.000	49.020
38	PV	0.000	0.000	0.000	92.593	0.000
39	PV	0.000	0.000	0.000	38.226	0.000
40	PV	0.000	0.000	0.000	84.746	0.000
41	J	0.000	0.000	0.000	0.000	113.122
42	PV	0.000	1.834	0.000	19.015	0.000
43	PV	0.000	22.200	0.000	0.413	0.972
44	PV	0.000	14.617	0.000	0.800	3.266
45	PM	2.019	0.000	5.143	11.719	0.000
46	PP	7.157	0.000	28.346	0.000	0.000
47	PK	21.818	0.000	5.151	0.000	0.000
48	J	0.000	0.000	0.000	0.000	71.633
49	PM	0.000	57.208	0.000	0.000	0.000
50	PV	0.000	0.000	0.000	40.388	0.000

Bayesiano; $x_{O_i}=\{Ray,t,R,G,B\}$						
n	Tipo	PK	PM	PP	PV	J
1	PP	1.000	0.000	0.000	0.000	0.000
2	PV	0.000	0.000	0.000	1.000	0.000
3	PV	0.000	0.000	0.000	0.973	0.027
4	PK	1.000	0.000	0.000	0.000	0.000
5	PP	0.010	0.000	0.000	0.000	0.990
6	J	0.000	0.000	0.000	0.000	1.000
7	J	0.000	0.000	0.000	0.000	1.000
8	PM	1.000	0.000	0.000	0.000	0.000
9	J	0.000	0.000	0.000	0.000	1.000
10	PV	0.000	0.000	0.000	1.000	0.000
11	PV	0.000	0.000	0.000	1.000	0.000
12	J	0.000	0.000	0.000	0.000	1.000
13	PK	1.000	0.000	0.000	0.000	0.000
14	PV	0.000	0.000	0.000	1.000	0.000
15	PK	1.000	0.000	0.000	0.000	0.000
16	J	0.000	0.000	0.000	0.000	1.000
17	PM	0.998	0.000	0.000	0.000	0.002
18	PP	1.000	0.000	0.000	0.000	0.000
19	PV	0.000	0.000	0.000	1.000	0.000
20	PP	0.000	0.000	0.000	1.000	0.000
21	J	0.000	0.000	0.000	0.000	1.000
22	PM	1.000	0.000	0.000	0.000	0.000
23	J	0.000	0.000	0.000	0.000	1.000
24	PK	1.000	0.000	0.000	0.000	0.000
25	PV	0.000	0.000	0.000	1.000	0.000
26	J	0.000	0.000	0.000	0.000	1.000
27	PK	1.000	0.000	0.000	0.000	0.000
28	PV	0.000	0.000	0.000	1.000	0.000
29	J	0.000	0.000	0.000	0.000	1.000
30	PK	0.000	0.000	0.000	1.000	0.000
31	J	0.000	0.000	0.000	0.000	1.000
32	J	0.000	0.000	0.000	0.000	1.000
33	PP	1.000	0.000	0.000	0.000	0.000
34	PK	1.000	0.000	0.000	0.000	0.000
35	PM	1.000	0.000	0.000	0.000	0.000
36	PP	1.000	0.000	0.000	0.000	0.000
37	J	0.000	0.000	0.000	0.000	1.000
38	PV	0.000	0.000	0.000	1.000	0.000
39	PV	0.000	0.000	0.000	1.000	0.000
40	PV	0.000	0.000	0.000	1.000	0.000
41	J	0.000	0.000	0.000	0.000	1.000
42	PV	0.000	0.000	0.000	1.000	0.000
43	PV	1.000	0.000	0.000	0.000	0.000
44	PV	0.000	0.000	0.000	0.000	1.000
45	PM	0.003	0.000	0.000	0.997	0.000
46	PP	0.000	0.000	0.000	0.000	1.000
47	PK	1.000	0.000	0.000	0.000	0.000
48	J	0.000	0.000	0.000	0.000	1.000
49	PM	1.000	0.000	0.000	0.000	0.000
50	PV	0.000	0.000	0.000	1.000	0.000

7.3.2. VECTOR $x_{A_i} = \{ColorA, t, r_c\}$

Distancias euclídeas; $x_{A_i}=\{eRay,t,gris\}$						
n	Tipo	PK	PM	PP	PV	J
1	PP	0.138	0.245	0.067	0.091	0.459
2	PV	0.182	0.252	0.136	0.033	0.397
3	PV	0.066	0.195	0.045	0.215	0.479
4	PK	0.051	0.086	0.142	0.336	0.384
5	PP	0.052	0.102	0.133	0.333	0.380
6	J	0.170	0.125	0.215	0.350	0.140
7	J	0.195	0.136	0.240	0.361	0.068
8	PM	0.019	0.137	0.101	0.298	0.445
9	J	0.215	0.170	0.245	0.322	0.048
10	PV	0.190	0.232	0.162	0.102	0.314
11	PV	0.184	0.247	0.144	0.053	0.373
12	J	0.214	0.156	0.253	0.349	0.028
13	PK	0.093	0.221	0.024	0.177	0.486
14	PV	0.188	0.237	0.155	0.082	0.338
15	PK	0.057	0.194	0.037	0.231	0.481
16	J	0.166	0.115	0.216	0.356	0.148
17	PM	0.174	0.086	0.232	0.356	0.152
18	PP	0.041	0.181	0.053	0.255	0.470
19	PV	0.173	0.260	0.118	0.013	0.436
20	PP	0.190	0.231	0.163	0.105	0.311
21	J	0.167	0.142	0.202	0.329	0.160
22	PM	0.083	0.049	1698.000	0.356	0.343
23	J	0.213	0.166	0.244	0.320	0.056
24	PK	0.166	0.071	0.229	0.368	0.167
25	PV	0.052	0.177	0.063	0.238	0.470
26	J	0.212	0.174	0.238	0.302	0.075
27	PK	0.005	0.150	0.092	0.299	0.454
28	PV	0.152	0.249	0.091	0.055	0.453
29	J	0.156	0.232	0.111	0.137	0.364
30	PK	0.194	0.220	0.177	0.140	0.269
31	J	0.209	0.160	0.246	0.349	0.036
32	J	0.209	0.172	0.233	0.290	0.095
33	PP	0.037	0.175	0.060	0.265	0.463
34	PK	0.012	0.151	0.091	0.293	0.452
35	PM	0.180	0.097	0.235	0.353	0.134
36	PP	0.047	0.184	0.048	0.247	0.473
37	J	0.212	0.167	0.224	0.319	0.057
38	PV	0.170	0.258	0.113	0.018	0.441
39	PV	0.186	0.244	0.148	0.064	0.359
40	PV	0.181	0.254	0.134	0.028	0.403
41	J	0.216	0.166	0.251	0.338	0.083
42	PV	0.089	0.214	0.032	0.180	0.485
43	PV	0.088	0.043	0.174	0.358	0.337
44	PV	0.116	0.023	0.197	0.366	0.299
45	PM	0.113	0.229	0.045	0.136	0.477
46	PP	0.117	0.235	0.038	0.142	0.469
47	PK	0.055	0.193	0.041	0.234	0.478
48	J	0.214	0.163	0.248	0.332	0.043
49	PM	0.143	0.032	0.216	0.377	0.232
50	PV	0.186	0.242	0.149	0.067	0.356

k-NN=25; $x_{A_i}=\{eRay,t,gris\}$						
n	Tipo	PK	PM	PP	PV	J
1	PP	8	0	14	3	0
2	PV	0	0	1	24	0
3	PV	11	0	9	5	0
4	PK	8	13	4	0	0
5	PP	6	11	8	0	0
6	J	5	0	0	0	20
7	J	0	0	0	0	25
8	PM	7	7	7	4	0
9	J	0	0	0	0	25
10	PV	0	0	0	25	0
11	PV	0	0	0	25	0
12	J	0	0	0	0	25
13	PK	12	0	11	2	0
14	PV	0	0	0	25	0
15	PK	10	0	13	2	0
16	J	6	1	0	0	18
17	PM	6	19	0	0	0
18	PP	10	0	15	0	0
19	PV	1	0	4	20	0
20	PP	0	0	0	25	0
21	J	3	0	1	0	21
22	PM	8	16	1	0	0
23	J	0	0	0	0	25
24	PK	6	19	0	0	0
25	PV	11	1	6	7	0
26	J	0	0	0	0	25
27	PK	8	3	11	3	0
28	PV	6	0	7	12	0
29	J	4	0	21	0	0
30	PK	0	0	0	25	0
31	J	0	0	0	0	25
32	J	0	0	0	0	25
33	PP	9	1	15	0	0
34	PK	8	2	11	4	0
35	PM	6	19	0	0	0
36	PP	11	0	13	1	0
37	J	0	0	0	0	25
38	PV	1	0	4	20	0
39	PV	0	0	0	25	0
40	PV	0	0	2	23	0
41	J	0	0	0	0	25
42	PV	12	0	8	5	0
43	PV	9	15	1	0	0
44	PV	6	19	0	0	0
45	PM	9	0	11	5	0
46	PP	6	0	19	0	0
47	PK	11	0	12	2	0
48	J	0	0	0	0	25
49	PM	6	19	0	0	0
50	PV	0	0	0	25	0

Híbrido kNN=25; $x_{Ai}=\{eRay,t,gris\}$						
n	Tipo	PK	PM	PP	PV	J
1	PP	6.898	0.000	13.065	2.727	0.000
2	PV	0.000	0.000	0.864	23.215	0.000
3	PV	10.279	0.000	8.594	3.924	0.000
4	PK	7.591	11.877	3.433	0.000	0.000
5	PP	5.690	9.883	6.934	0.000	0.000
6	J	4.151	0.000	0.000	0.000	17.193
7	J	0.000	0.000	0.000	0.000	23.300
8	PM	6.867	6.038	6.291	2.810	0.000
9	J	0.000	0.000	0.000	0.000	23.804
10	PV	0.000	0.000	0.000	22.456	0.000
11	PV	0.000	0.000	0.000	23.683	0.000
12	J	0.000	0.000	0.000	0.000	24.289
13	PK	10.885	0.000	10.740	1.647	0.000
14	PV	0.000	0.000	0.000	30.357	0.000
15	PK	9.428	0.000	12.518	1.538	0.000
16	J	5.006	0.885	0.000	0.000	15.344
17	PM	4.956	17.365	0.000	0.000	0.000
18	PP	9.586	0.000	14.205	0.000	0.000
19	PV	0.827	0.000	3.527	19.737	0.000
20	PP	0.000	0.000	0.000	22.388	0.000
21	J	2.500	0.000	0.798	0.000	17.631
22	PM	7.335	15.219	0.830	0.000	0.000
23	J	0.000	0.000	0.000	0.000	23.601
24	PK	5.006	17.659	0.000	0.000	0.000
25	PV	10.427	0.823	5.620	5.333	0.000
26	J	0.000	0.000	0.000	0.000	23.117
27	PK	7.957	2.551	9.985	2.104	0.000
28	PV	5.090	0.000	6.363	11.339	0.000
29	J	3.376	0.000	18.671	0.000	0.000
30	PK	0.000	0.000	0.000	21.495	0.000
31	J	0.000	0.000	0.000	0.000	24.109
32	J	0.000	0.000	0.000	0.000	22.614
33	PP	8.666	0.825	14.094	0.000	0.000
34	PK	7.907	1.698	9.947	2.826	0.000
35	PM	4.917	17.151	0.000	0.000	0.000
36	PP	10.483	0.000	12.373	0.753	0.000
37	J	0.000	0.000	0.000	0.000	23.570
38	PV	0.590	0.000	3.528	109.815	0.000
39	PV	0.000	0.000	0.000	23.390	0.000
40	PV	0.000	0.000	1.731	22.368	0.000
41	J	0.000	0.000	0.000	0.000	24.292
42	PV	10.932	0.000	7.742	4.099	0.000
43	PV	8.209	14.352	0.826	0.000	0.000
44	PV	5.303	18.571	0.000	0.000	0.000
45	PM	7.988	0.000	10.502	4.320	0.000
46	PP	5.299	0.000	18.288	0.000	0.000
47	PK	10.398	0.000	11.512	1.533	0.000
48	J	0.000	0.000	0.000	0.000	23.922
49	PM	5.143	18.398	0.000	0.000	0.000
50	PV	0.000	0.000	0.000	23.321	0.000

Híbrido alternativo kNN=25; $x_{Ai}=\{eRay,t,gris\}$						
n	Tipo	PK	PM	PP	PV	J
1	PP	5.805	0.000	20.967	3.294	0.000
2	PV	0.000	0.000	0.734	73.355	0.000
3	PV	16.970	0.000	19.970	2.322	0.000
4	PK	15.645	15.047	2.823	0.000	0.000
5	PP	11.606	10.828	6.003		
6	J	2.945	0.000	0.000	0.000	14.252
7	J	0.000	0.000	0.000	0.000	36.764
8	PM	36.919	5.094	6.910	1.344	0.000
9	J	0.000	0.000	0.000	0.000	52.268
10	PV	0.000	0.000	0.000	24.568	0.000
11	PV	0.000	0.000	0.000	47.446	0.000
12	J	0.000	0.000	0.000	0.000	87.955
13	PK	12.919	0.000	46.449	1.133	0.000
14	PV	0.000	0.000	0.000	22.941	0.000
15	PK	17.748	0.000	35.027	0.865	0.000
16	J	3.620	0.867	0.000	0.000	12.200
17	PM	0.000	3.448	22.085	0.000	0.000
18	PP	24.143	.	28.298	0.000	0.000
19	PV	0.577	0.000	3.382	152.018	0.000
20	PP	0.000	0.000	0.000	23.923	0.000
21	J	1.800	0.000	0.494	0.000	13.091
22	PM	9.629	32.792	0.589	0.000	0.000
23	J	0.000	0.000	0.000	0.000	44.658
24	PK	3.621	26.912	0.000	0.000	0.000
25	PV	21.110	0.565	9.480	2.939	0.000
26	J	0.000	0.000	0.000	0.000	33.193
27	PK	148.560	2.002	11.924	1.004	0.000
28	PV	3.954	0.000	7.693	21.778	0.000
29	J	2.566	0.000	18.936	0.000	0.000
30	PK	0.000	0.000	0.000	17.832	0.000
31	J	0.000	0.000	0.000	0.000	70.176
32	J	0.000	0.000	0.000	0.000	26.195
33	PP	24.282	0.573	24.839	0.000	0.000
34	PK	68.762	1.324	12.036	1.363	0.000
35	PM	3.325	19.524	0.000	0.000	0.000
36	PP	23.415	0.000	26.944	0.405	0.000
37	J	0.000	0.000	0.000	0.000	43.718
38	PV	0.831	0.000	3.546	19.636	0.000
39	PV	0.000	0.000	0.000	38.826	0.000
40	PV	0.000	0.000	1.489	86.710	0.000
41	J	0.000	0.000	0.000	0.000	88.209
42	PV	13.487	0.000	24.765	2.774	0.000
43	PV	10.239	34.722	0.575	0.000	0.000
44	PV	5.168	84.071	0.000	0.000	0.000
45	PM	8.000	0.000	24.283	3.674	0.000
46	PP	5.139	0.000	50.698	0.000	0.000
47	PK	20.100	0.000	29.520	0.856	0.000
48	J	0.000	0.000	0.000	0.000	57.975
49	PM	4.199	59.918	0.000	0.000	0.000
50	PV	0.000	0.000	0.000	37.224	0.000

Bayesiano; $x_{Ai}=\{eRay,t,gris\}$						
n	Tipo	PK	PM	PP	PV	J
1	PP	0.000	0.000	1.000	0.000	0.000
2	PV	0.000	0.000	0.000	1.000	0.000
3	PV	0.000	0.001	0.000	0.999	0.000
4	PK	0.752	0.235	0.009	0.005	0.000
5	PP	0.000	0.000	1.000	0.000	0.000
6	J	0.000	0.000	0.000	0.000	1.000
7	J	0.000	0.000	0.000	0.000	1.000
8	PM	0.000	0.679	0.001	0.321	0.000
9	J	0.000	0.000	0.000	0.000	1.000
10	PV	0.000	0.000	0.000	1.000	0.000
11	PV	0.000	0.000	0.000	1.000	0.000
12	J	0.000	0.000	0.000	0.000	1.000
13	PK	1.000	0.000	0.000	0.000	0.000
14	PV	0.000	0.000	0.000	1.000	0.000
15	PK	0.787	0.006	0.205	0.002	0.000
16	J	0.000	0.000	0.000	0.000	1.000
17	PM	0.000	0.957	0.000	0.043	0.000
18	PP	0.002	0.000	0.998	0.000	0.000
19	PV	0.000	0.000	0.000	1.000	0.000
20	PP	0.001	0.000	0.000	0.999	0.000
21	J	0.000	0.000	0.000	0.000	1.000
22	PM	0.000	0.993	0.000	0.007	0.000
23	J	0.000	0.000	0.000	0.000	1.000
24	PK	1.000	0.000	0.000	0.000	0.000
25	PV	0.000	0.007	0.000	0.993	0.000
26	J	0.000	0.000	0.000	0.000	1.000
27	PK	1.000	0.000	0.000	0.000	0.000
28	PV	0.000	0.000	0.000	1.000	0.000
29	J	0.000	0.000	0.000	0.000	1.000
30	PK	0.000	0.000	0.000	1.000	0.000
31	J	0.000	0.000	0.000	0.000	1.000
32	J	0.000	0.000	0.000	0.000	1.000
33	PP	0.000	0.000	1.000	0.000	0.000
34	PK	1.000	0.000	0.000	0.000	0.000
35	PM	0.000	0.998	0.000	0.002	0.000
36	PP	0.074	0.022	0.902	0.002	0.000
37	J	0.000	0.000	0.000	0.000	1.000
38	PV	0.000	0.000	0.000	1.000	0.000
39	PV	0.000	0.000	0.000	1.000	0.000
40	PV	0.000	0.000	0.000	1.000	0.000
41	J	0.000	0.000	0.000	0.000	1.000
42	PV	0.000	0.022	0.000	0.978	0.000
43	PV	0.000	0.992	0.000	0.008	0.000
44	PV	0.000	0.000	0.000	1.000	0.000
45	PM	0.000	0.000	0.000	0.998	0.000
46	PP	0.000	0.000	1.000	0.000	0.000
47	PK	1.000	0.000	0.000	0.000	0.000
48	J	0.000	0.000	0.000	0.000	1.000
49	PM	0.042	0.957	0.000	0.001	0.000
50	PV	0.000	0.000	0.000	1.000	0.000

7.3.3. VECTOR $x_{B_i} = \{ColorB, t, r_c\}$

Distancias euclídeas; $x_{Bi}=\{eRay,t,RGRB\}$						
n	Tipo	PK	PM	PP	PV	J
1	PP	0.012	0.204	0.055	0.332	0.398
2	PV	0.352	0.147	0.442	0.011	0.068
3	PV	0.333	0.113	0.409	0.032	0.113
4	PK	0.011	0.191	0.080	0.324	0.394
5	PP	0.108	0.214	0.071	0.285	0.321
6	J	0.347	0.176	0.406	0.066	0.004
7	J	0.350	0.169	0.412	0.054	0.015
8	PM	0.233	0.013	0.318	0.175	0.261
9	J	0.346	0.176	0.405	0.068	0.006
10	PV	0.352	0.148	0.422	0.012	0.066
11	PV	0.287	0.052	0.369	0.104	0.188
12	J	0.349	0.153	0.417	0.033	0.048
13	PK	0.051	0.159	0.123	0.298	0.370
14	PV	0.331	0.111	0.407	0.035	0.115
15	PK	0.036	0.214	0.026	0.332	0.393
16	J	0.347	0.176	0.406	0.065	0.007
17	PM	0.222	0.022	0.306	0.183	0.268
18	PP	0.066	0.147	0.139	0.288	0.361
19	PV	0.346	0.166	0.409	0.046	0.033
20	PP	0.042	0.217	0.018	0.033	0.392
21	J	0.348	0.176	0.408	0.066	0.001
22	PM	0.210	0.031	0.293	0.191	0.275
23	J	0.345	0.177	0.403	0.069	0.007
24	PK	0.047	0.218	0.012	0.332	0.391
25	PV	0.348	0.160	0.414	0.035	0.043
26	J	0.334	0.179	0.388	0.079	0.020
27	PK	0.015	0.188	0.084	0.322	0.392
28	PV	0.348	0.162	0.412	0.038	0.039
29	J	0.347	0.176	0.407	0.066	0.004
30	PK	0.318	0.092	0.395	0.057	0.138
31	J	0.343	0.144	0.412	0.039	0.062
32	J	0.350	0.166	0.413	0.049	0.021
33	PP	0.061	0.222	0.006	0.328	0.383
34	PK	0.064	0.221	0.010	0.326	0.380
35	PM	0.194	0.044	0.276	0.202	0.285
36	PP	0.138	0.210	0.114	0.257	0.281
37	J	0.339	0.178	0.395	0.074	0.014
38	PV	0.349	0.158	0.415	0.031	0.047
39	PV	0.339	0.121	0.414	0.023	0.103
40	PV	0.350	0.156	0.417	0.027	0.051
41	J	0.344	0.177	0.402	0.070	0.007
42	PV	0.353	0.143	0.425	0.004	0.075
43	PV	0.350	0.158	0.416	0.030	0.046
44	PV	0.351	0.151	0.420	0.018	0.061
45	PM	0.260	0.016	0.345	0.147	0.233
46	PP	0.088	0.130	0.162	0.273	0.348
47	PK	0.193	0.045	0.275	0.203	0.285
48	J	0.344	0.177	0.401	0.069	0.009
49	PM	0.290	0.055	0.371	0.101	0.184
50	PV	0.353	0.143	0.425	0.005	0.075

k-NN=25; $x_{Bi}=\{eRay,t,RGRB\}$						
n	Tipo	PK	PM	PP	PV	J
1	PP	11	0	14	0	0
2	PV	0	0	0	25	0
3	PV	0	0	0	25	0
4	PK	10	0	15	0	0
5	PP	5	0	20	0	0
6	J	0	0	0	0	25
7	J	0	0	0	0	25
8	PM	0	25	0	0	0
9	J	0	0	0	0	25
10	PV	0	0	0	25	0
11	PV	0	24	0	1	0
12	J	0	0	0	19	6
13	PK	18	0	7	0	0
14	PV	0	0	0	25	0
15	PK	14	0	11	0	0
16	J	0	0	0	0	25
17	PM	0	25	0	0	0
18	PP	20	1	4	0	0
19	PV	0	0	0	15	10
20	PP	14	0	11	0	0
21	J	0	0	0	0	25
22	PM	0	25	0	0	0
23	J	0	0	0	0	25
24	PK	15	0	10	0	0
25	PV	0	0	0	24	1
26	J	0	0	0	0	25
27	PK	10	0	15	0	0
28	PV	0	0	0	21	4
29	J	0	0	0	0	25
30	PK	0	7	0	18	0
31	J	0	0	0	19	6
32	J	0	0	0	0	25
33	PP	12	0	13	0	0
34	PK	12	0	13	0	0
35	PM	2	23	0	0	0
36	PP	5	0	20	0	0
37	J	0	0	0	0	25
38	PV	0	0	0	25	0
39	PV	0	0	0	25	0
40	PV	0	0	0	25	0
41	J	0	0	0	0	25
42	PV	0	0	0	25	0
43	PV	0	0	0	25	0
44	PV	0	0	0	25	0
45	PM	0	25	0	0	0
46	PP	20	2	3	0	0
47	PK	2	23	0	0	0
48	J	0	0	0	0	25
49	PM	0	23	0	2	0
50	PV	0	0	0	25	0

Híbrido kNN=25; xBi={eRay,t,RGRB}						
n	Tipo	PK	PM	PP	PV	J
1	PP	10.874	0.000	13.227	0.000	0.000
2	PV	0.000	0.000	0.000	24.729	0.000
3	PV	0.000	0.000	0.000	24.196	0.000
4	PK	9.894	0.000	13.799	0.000	0.000
5	PP	4.460	0.000	18.574	0.000	0.000
6	J	0.000	0.000	0.000	0.000	24.889
7	J	0.000	0.000	0.000	0.000	24.617
8	PM	0.000	24.687	0.000	0.000	0.000
9	J	0.000	0.000	0.000	0.000	24.861
10	PV	0.000	0.000	0.000	24.689	0.000
11	PV	0.000	22.756	0.000	0.896	0.000
12	J	0.000	0.000	0.000	18.375	5.715
13	PK	17.090	0.000	6.140	0.000	0.000
14	PV	0.000	0.000	0.000	24.138	0.000
15	PK	13.500	0.000	10.717	0.000	0.000
16	J	0.000	0.000	0.000	0.000	24.822
17	PM	0.000	24.459	0.000	0.000	0.000
18	PP	18.688	0.853	3.448	0.000	0.000
19	PV	0.000	0.000	0.000	14.309	9.670
20	PP	13.406	0.000	10.805	0.000	0.000
21	J	0.000	0.000	0.000	0.000	24.966
22	PM	0.000	24.229	0.000	0.000	0.000
23	J	0.000	0.000	0.000	0.000	24.830
24	PK	14.292	0.000	9.881	0.000	0.000
25	PV	0.000	0.000	0.000	23.164	0.957
26	J	0.000	0.000	0.000	0.000	24.497
27	PK	9.854	0.000	13.735	0.000	0.000
28	PV	0.000	0.000	0.000	20.176	3.843
29	J	0.000	0.000	0.000	0.000	24.893
30	PK	0.000	6.355	0.000	16.969	0.000
31	J	0.000	0.000	0.000	18.252	5.627
32	J	0.000	0.000	0.000	0.000	24.467
33	PP	11.265	0.000	12.923	0.000	0.000
34	PK	11.235	0.000	12.870	0.000	0.000
35	PM	1.613	21.983	0.000	0.000	0.000
36	PP	4.308	0.000	17.725	0.000	0.000
37	J	0.000	0.000	0.000	0.000	24.650
38	PV	0.000	0.000	0.000	24.220	0.000
39	PV	0.000	0.000	0.000	24.427	0.000
40	PV	0.000	0.000	0.000	24.328	0.000
41	J	0.000	0.000	0.000	0.000	24.838
42	PV	0.000	0.000	0.000	24.899	0.000
43	PV	0.000	0.000	0.000	24.243	0.000
44	PV	0.000	0.000	0.000	24.548	0.000
45	PM	0.000	24.610	0.000	0.000	0.000
46	PP	18.247	1.741	2.513	0.000	0.000
47	PK	1.615	21.960	0.000	0.000	0.000
48	J	0.000	0.000	0.000	0.000	24.783
49	PM	0.000	21.733	0.000	1.799	0.000
50	PV	0.000	0.000	0.000	24.888	0.000

Híbrido alternativo kNN=25; xBi={eRay,t,RGRB}						
n	Tipo	PK	PM	PP	PV	J
1	PP	95.819	0.000	25.351	0.000	0.000
2	PV	0.000	0.000	0.000	230.482	0.000
3	PV	0.000	0.000	0.000	77.771	0.000
4	PK	93.922	0.000	18.728	0.000	0.000
5	PP	4.629	0.000	28.043	0.000	0.000
6	J	0.000	0.000	0.000	0.000	563.446
7	J	0.000	0.000	0.000	0.000	163.188
8	PM	0.000	199.533	0.000	0.000	0.000
9	J	0.000	0.000	0.000	0.000	450.288
10	PV	0.000	0.000	0.000	201.045	0.000
11	PV	0.000	462.818	0.000	9.600	0.000
12	J	0.000	0.000	0.000	57.755	12.631
13	PK	35.584	0.000	5.698	0.000	0.000
14	PV	0.000	0.000	0.000	72.539	0.000
15	PK	39.175	0.000	42.706	0.000	0.000
16	J	0.000	0.000	0.000	0.000	351.509
17	PM	0.000	115.591	0.000	0.000	0.000
18	PP	30.492	0.680	2.882	0.000	0.000
19	PV	0.000	0.000	0.000	32.553	30.317
20	PP	33.019	0.000	62.147	0.000	0.000
21	J	0.000	0.000	0.000	0.000	1833.500
22	PM	0.000	81.057	0.000	0.000	0.000
23	J	0.000	0.000	0.000	0.000	366.636
24	PK	31.757	0.000	83.773	0.000	0.000
25	PV	0.000	0.000	0.000	68.860	2.304
26	J	0.000	0.000	0.000	0.000	124.139
27	PK	68.659	0.000	17.784	0.000	0.000
28	PV	0.000	0.000	0.000	53.499	10.169
29	J	0.000	0.000	0.000	0.000	581.395
30	PK	0.000	7.592	0.000	31.414	0.000
31	J	0.000	0.000	0.000	48.249	9.646
32	J	0.000	0.000	0.000	0.000	117.179
33	PP	19.578	0.000	218.743	0.000	0.000
34	PK	18.823	0.000	130.129	0.000	0.000
35	PM	1.033	52.015	0.000	0.000	0.000
36	PP	3.614	0.000	17.583	0.000	0.000
37	J	0.000	0.000	0.000	0.000	178.755
38	PV	0.000	0.000	0.000	80.167	0.000
39	PV	0.000	0.000	0.000	109.006	0.000
40	PV	0.000	0.000	0.000	93.046	0.000
41	J	0.000	0.000	0.000	0.000	384.875
42	PV	0.000	0.000	0.000	621.313	0.000
43	PV	0.000	0.000	0.000	82.508	0.000
44	PV	0.000	0.000	0.000	138.122	0.000
45	PM	0.000	160.256	0.000	0.000	0.000
46	PP	22.813	1.545	1.849	0.000	0.000
47	PK	1.039	50.870	0.000	0.000	0.000
48	J	0.000	0.000	0.000	0.000	287.319
49	PM	0.000	41.759	0.000	1.988	0.000
50	PV	0.000	0.000	0.000	560.118	0.000

Bayesiano; $x_{Bi}=\{eRay,t,RGRB\}$						
n	Tipo	PK	PM	PP	PV	J
1	PP	0.000	0.000	1.000	0.000	0.000
2	PV	0.000	0.000	0.000	1.000	0.000
3	PV	0.000	0.000	0.000	1.000	0.000
4	PK	0.979	0.000	0.021	0.000	0.001
5	PP	0.000	0.000	1.000	0.000	0.000
6	J	0.000	0.000	0.000	0.000	1.000
7	J	0.000	0.000	0.000	0.000	1.000
8	PM	0.000	1.000	0.000	0.000	0.000
9	J	0.000	0.000	0.000	0.000	1.000
10	PV	0.000	0.000	0.000	1.000	0.000
11	PV	0.000	0.000	0.000	1.000	0.000
12	J	0.000	0.000	0.000	0.000	1.000
13	PK	1.000	0.000	0.000	0.000	0.000
14	PV	0.000	0.000	0.000	1.000	0.000
15	PK	0.853	0.000	0.147	0.000	0.000
16	J	0.000	0.000	0.000	0.000	1.000
17	PM	0.000	1.000	0.000	0.000	0.000
18	PP	0.008	0.000	0.992	0.000	1.000
19	PV	0.000	0.000	0.000	1.000	0.000
20	PP	0.480	0.000	0.520	0.000	0.000
21	J	0.000	0.000	0.000	0.000	1.000
22	PM	0.000	1.000	0.000	0.000	0.000
23	J	0.000	0.000	0.000	0.000	1.000
24	PK	1.000	0.000	0.000	0.000	0.000
25	PV	0.000	0.000	0.000	1.000	0.000
26	J	0.000	0.000	0.000	0.000	1.000
27	PK	1.000	0.000	0.000	0.000	0.000
28	PV	0.000	0.000	0.000	1.000	0.000
29	J	0.000	0.000	0.000	0.000	1.000
30	PK	0.586	0.189	0.000	0.225	0.000
31	J	0.000	0.000	0.000	0.000	1.000
32	J	0.000	0.000	0.000	0.000	1.000
33	PP	0.000	0.000	1.000	0.000	0.000
34	PK	1.000	0.000	0.000	0.000	0.000
35	PM	0.000	1.000	0.000	0.000	0.000
36	PP	0.002	0.000	0.998	0.000	0.000
37	J	0.000	0.000	0.000	0.000	1.000
38	PV	0.000	0.000	0.000	1.000	0.000
39	PV	0.000	0.054	0.000	0.947	0.000
40	PV	0.000	0.000	0.000	1.000	0.000
41	J	0.000	0.000	0.000	0.000	1.000
42	PV	0.000	0.001	0.000	0.999	0.000
43	PV	0.000	0.001	0.000	0.999	0.000
44	PV	0.000	0.000	0.000	1.000	0.000
45	PM	0.000	1.000	0.000	0.000	0.000
46	PP	0.000	0.000	1.000	0.000	0.000
47	PK	1.000	0.000	0.000	0.000	0.000
48	J	0.000	0.000	0.000	0.000	1.000
49	PM	0.001	0.998	0.000	0.001	0.000
50	PV	0.000	0.000	0.000	1.000	0.000

7.4. FUNCIÓN DE CLASIFICACIÓN

```
%%% Clasificadores %%%
function [dEs, nnKs, dEhib, dEhib2, Pbs]=clasificadores(vC, NN)
%% Carga de datos
% Se cargan las matrices correspondientes a la base de datos para cada
% especie, donde cada columna es una característica.
load('PK.mat');
load('PM.mat');
load('PP.mat');
load('PV.mat');
load('J.mat');

%% Cálculos previos
% Cálculo de centros de clúster/valor promedio
MK=[sum(PK(:,1)) sum(PK(:,2)) sum(PK(:,3))]/size(PK,1);
MM=[sum(PM(:,1)) sum(PM(:,2)) sum(PM(:,3))]/size(PM,1);
MP=[sum(PP(:,1)) sum(PP(:,2)) sum(PP(:,3))]/size(PP,1);
MV=[sum(PV(:,1)) sum(PV(:,2)) sum(PV(:,3))]/size(PV,1);
MJ=[sum(J(:,1)) sum(J(:,2)) sum(J(:,3))]/size(J,1);
% Promedios
medX=[MK(1) MM(1) MP(1) MV(1) MJ(1)]; % X -> Rayado
medY=[MK(2) MM(2) MP(2) MV(2) MJ(2)]; % Y -> Tamaño
medZ=[MK(3) MM(3) MP(3) MV(3) MJ(3)]; % Z -> Color

% Cálculo de varianza
MKv=[sum(PK(:,1).^2) sum(PK(:,2).^2) sum(PK(:,3).^2)]/size(PK,1);
MMv=[sum(PM(:,1).^2) sum(PM(:,2).^2) sum(PM(:,3).^2)]/size(PM,1);
MPv=[sum(PP(:,1).^2) sum(PP(:,2).^2) sum(PP(:,3).^2)]/size(PP,1);
MVv=[sum(PV(:,1).^2) sum(PV(:,2).^2) sum(PV(:,3).^2)]/size(PV,1);
MJv=[sum(J(:,1).^2) sum(J(:,2).^2) sum(J(:,3).^2)]/size(J,1);

% Varianzas
varX=[(MKv(1)-MK(1)^2) (MMv(1)-MM(1)^2) (MPv(1)-MP(1)^2) (MVv(1)-
MV(1)^2) (MJv(1)-MJ(1)^2)]; % X -> Rayado
varY=[(MKv(2)-MK(2)^2) (MMv(2)-MM(2)^2) (MPv(2)-MP(2)^2) (MVv(2)-
MV(2)^2) (MJv(2)-MJ(2)^2)]; % Y -> Tamaño
varZ=[(MKv(3)-MK(3)^2) (MMv(3)-MM(3)^2) (MPv(3)-MP(3)^2) (MVv(3)-
MV(3)^2) (MJv(3)-MJ(3)^2)]; % Z -> Color

%% Clasificador por distancia euclídea
demK=0;
demM=0;
demP=0;
demV=0;
demJ=0;
for i=1:3
    demK=demK+(vC(i)-MK(i))^2;
    demM=demM+(vC(i)-MM(i))^2;
    demP=demP+(vC(i)-MP(i))^2;
```

```

        demV=demV+(vC(i)-MV(i))^2;
        demJ=demJ+(vC(i)-MJ(i))^2;
end
demK=sqrt(demK);
demM=sqrt(demM);
demP=sqrt(demP);
demV=sqrt(demV);
demJ=sqrt(demJ);
dEs=[demK demM demP demV demJ]/(demK+demM+demP+demV+demJ); % Distancia
normalizada

%% Clasificador por vecinos cercanos
nnK=0;
nnM=0;
nnP=0;
nnV=0;
nnJ=0;

% Vectores con valores altos y diferentes para evitar errores en
igualdades
kNNeucK=1000*ones(1,NN);
for aux=1:NN
    kNNeucK(aux)=kNNeucK(aux)+aux;
end
kNNeucM=kNNeucK;
kNNeucP=kNNeucK;
kNNeucV=kNNeucK;
kNNeucJ=kNNeucK;

%PK
for u=1:size(PK,1) % Para cada punto
    dekK=sqrt((vC(1)-PK(u,1))^2+(vC(2)-PK(u,2))^2+(vC(3)-PK(u,3))^2);
    if dekK<max(kNNeucK) % si es menor que la máxima distancia
        posMax=kNNeucK==max(kNNeucK);
        kNNeucK(posMax)=dekK; % sustituye la posición con el nuevo
valor
    end
end
%PM
for u=1:size(PM,1) % Para cada punto
    dekM=sqrt((vC(1)-PM(u,1))^2+(vC(2)-PM(u,2))^2+(vC(3)-PM(u,3))^2);
    if dekM<max(kNNeucM) % si es menor que la máxima distancia
        posMax=kNNeucM==max(kNNeucM);
        kNNeucM(posMax)=dekM; % sustituye la posición con el nuevo
valor
    end
end
%PP
for u=1:size(PP,1) % Para cada punto
    dekP=sqrt((vC(1)-PP(u,1))^2+(vC(2)-PP(u,2))^2+(vC(3)-PP(u,3))^2);
    if dekP<max(kNNeucP) % si es menor que la máxima distancia

```

```

        posMax=kNNeucP==max(kNNeucP);
        kNNeucP(posMax)=dekP; % sustituye la posición con el nuevo
valor
    end
end
%PV
for u=1:size(PV,1) % Para cada punto
    dekV=sqrt((vC(1)-PV(u,1))^2+(vC(2)-PV(u,2))^2+(vC(3)-PV(u,3))^2);
    if dekV<max(kNNeucV) % si es menor que la máxima distancia
        posMax=kNNeucV==max(kNNeucV);
        kNNeucV(posMax)=dekV; % sustituye la posición con el nuevo
valor
    end
end
%J
for u=1:size(J,1) % Para cada punto
    dekJ=sqrt((vC(1)-J(u,1))^2+(vC(2)-J(u,2))^2+(vC(3)-J(u,3))^2);
    if dekJ<max(kNNeucJ) % si es menor que la máxima distancia
        posMax=kNNeucJ==max(kNNeucJ);
        kNNeucJ(posMax)=dekJ; % sustituye la posición con el nuevo
valor
    end
end

% Matriz con N distancias mínimas a cada tipo
Mdist=[kNNeucK; kNNeucM;kNNeucP; kNNeucV; kNNeucJ];
for w=1:NN
    [x, y]=find(Mdist==min(min(Mdist)));
    if x==1
        nnK=nnK+1;
    elseif x==2
        nnM=nnM+1;
    elseif x==3
        nnP=nnP+1;
    elseif x==4
        nnV=nnV+1;
    else
        nnJ=nnJ+1;
    end
    Mdist(x,y)=max(max(Mdist));
end
nnKs=[nnK nnM nnP nnV nnJ];

%% Híbrido
dEhib=[nnK*(1-dEs(1)) nnM*(1-dEs(2)) nnP*(1-dEs(3)) nnV*(1-dEs(4))
nnJ*(1-dEs(5))];

% Alternativa de híbrido
dEhib2=0.1*[nnK/dEs(1) nnM/dEs(2) nnP/dEs(3) nnV/dEs(4) nnJ/dEs(5)];

```

```

%% Bayesiano

% PK
PrK=(2*pi*varX(1))^(-0.5)*exp(-(vC(1)-medX(1))^2/(2*varX(1))); %
Rayado PK
PtK=(2*pi*varY(1))^(-0.5)*exp(-(vC(2)-medY(1))^2/(2*varY(1))); %
Tamaño PK
PcK=(2*pi*varZ(1))^(-0.5)*exp(-(vC(3)-medZ(1))^2/(2*varZ(1))); % Color
PK
% PM
PrM=(2*pi*varX(2))^(-0.5)*exp(-(vC(1)-medX(2))^2/(2*varX(2))); %
Rayado PM
PtM=(2*pi*varY(2))^(-0.5)*exp(-(vC(2)-medY(2))^2/(2*varY(2))); %
Tamaño PM
PcM=(2*pi*varZ(2))^(-0.5)*exp(-(vC(3)-medZ(2))^2/(2*varZ(2))); % Color
PM
% PP
PrP=(2*pi*varX(3))^(-0.5)*exp(-(vC(1)-medX(3))^2/(2*varX(3))); %
Rayado PP
PtP=(2*pi*varY(3))^(-0.5)*exp(-(vC(2)-medY(3))^2/(2*varY(3))); %
Tamaño PP
PcP=(2*pi*varZ(3))^(-0.5)*exp(-(vC(3)-medZ(3))^2/(2*varZ(3))); % Color
PP
% PV
PrV=(2*pi*varX(4))^(-0.5)*exp(-(vC(1)-medX(4))^2/(2*varX(4))); %
Rayado PV
PtV=(2*pi*varY(4))^(-0.5)*exp(-(vC(2)-medY(4))^2/(2*varY(4))); %
Tamaño PV
PcV=(2*pi*varZ(4))^(-0.5)*exp(-(vC(3)-medZ(4))^2/(2*varZ(4))); % Color
PV
% PJ
PrJ=(2*pi*varX(5))^(-0.5)*exp(-(vC(1)-medX(5))^2/(2*varX(5))); %
Rayado J
PtJ=(2*pi*varY(5))^(-0.5)*exp(-(vC(2)-medY(5))^2/(2*varY(5))); %
Tamaño J
PcJ=(2*pi*varZ(5))^(-0.5)*exp(-(vC(3)-medZ(5))^2/(2*varZ(5))); % Color
J

% Normalización
pNorm=0.2*(PrK*PtK*PcK+PrM*PtM*PcM+PrP*PtP*PcP+PrV*PtV*PcV+PrJ*PtJ*PcJ
);

% Probabilidades bayesianas de pertenencia a clase
PbK=0.2*PrK*PtK*PcK/pNorm;
PbM=0.2*PrM*PtM*PcM/pNorm;
PbP=0.2*PrP*PtP*PcP/pNorm;
PbV=0.2*PrV*PtV*PcV/pNorm;
PbJ=0.2*PrJ*PtJ*PcJ/pNorm;

Pbs=[PbK PbM PbP PbV PbJ];

```