



UNIVERSITAT ROVIRA i VIRGILI

Práctica Visión Por Computador

Clasificación de Imágenes

Tomás Cantón Cordeiro
Visión Por Computador

1. Especificaciones del problema a resolver	3
2. Diseño e Implementación	4
3. Evolución de una de las imágenes durante el preprocesamiento	11
4. Juego de Pruebas	14
5. BIBLIOGRAFÍA	15

1. Especificaciones del problema a resolver

El problema a resolver consiste en desarrollar un programa que realice la clasificación de imágenes de gusanos intestinales capturadas con un microscopio. El programa debe clasificar cada imagen en dos categorías: gusanos vivos y gusanos muertos, y también debe informar la cantidad de gusanos vivos y muertos presentes en cada imagen.

Los gusanos vivos se caracterizan por tener una forma curvilínea, mientras que los gusanos muertos tienen una forma rectilínea. Algunas imágenes contendrán gusanos vivos, otras gusanos muertos, y otras imágenes contendrán ambos. En este último caso, la imagen se clasificará en la categoría que tenga la mayoría de los gusanos.

El programa deberá recibir como parámetro las imágenes a clasificar, y retornar la precisión del programa. Además, se deberá indicar la cantidad de gusanos vivos y muertos encontrados.

2. Diseño e Implementación

Para iniciar la práctica primero se realizó un esquema de los posibles pasos a seguir. Dado que nuestro objetivo es clasificar gusanos, nuestra prioridad es obtener el contorno de los gusanos, se realizará el siguiente preprocesamiento.

1. Binarizar la imagen
2. Detección de contornos
3. Segmentación de la imagen
4. Clasificación de Objetos

Se realizó tanto binarización global como local, obteniendo resultados similares con mucho ruido y no aptos para continuar el proceso. Se decidió aplicar técnicas de filtrado y aumento de contraste antes de binarizar la imagen.

Inicialmente se aumentará el contraste de la imagen para visibilizar los objetos de la imagen más marcados frente al fondo microscópico. En este punto se realizó un histograma de la imagen para comprobar si se podía concretar los gusanos en algún rango de valores. Se observa que no hay discontinuidades notorias y por tanto no se pueden agrupar a los gusanos en un intervalo determinado.

Se aplica un filtro de suavizado de mediana que sirve para eliminar suciedad de la imagen (no se utilizó hasta versiones finales donde se observó una menor cantidad de ruido). Luego se binariza la imagen utilizando un lindar local y se limpia eliminando los objetos menores a 200 px obteniendo la siguiente imagen.

```
% Preprocesamiento

contrast = imadjust(cropped, [], [], 1.5);
filtered = medfilt2(contrast, [5 5]);
bin = ~imbinarize(filtered, 'adaptive', 'Sensitivity', 1);
clean = bwareaopen(bin, 200);
```

Fig 1. Pseudocódigo preprocesamiento

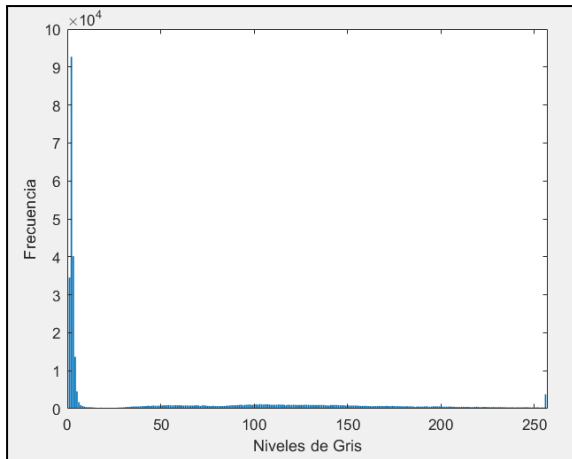
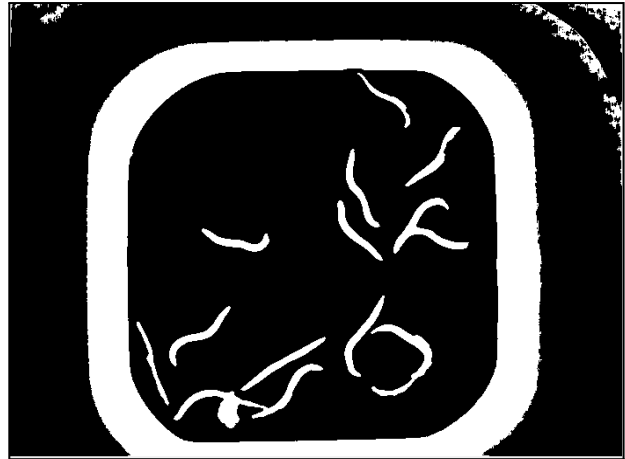


Fig 2. Histograma de la imagen



Img 2.1 Imagen bina izada después del preprocesamiento

Al realizar el primer preprocesamiento planteado, varios gusanos se fusionan en uno solo, además se detecta ruido en el fondo que se reconocerá como gusanos y dará lugar a interpretaciones incorrectas.

Para eliminar el ruido se realiza como paso inicial, un reescalado de la imagen que se recorta para cuadrar sólo los objetos de interés dentro de la imagen microscópica; luego se siguió con el preprocesamiento anterior. Como output se obtuvo la imagen anterior sin ruido exterior.

```
%Reajustar la imagen y recortar background
resized = imresize(img, 1.5);

[h, w] = size(img);
w_crop = round(w * 0.2);
h_crop = round(h * 0.05);
cropped = resized(h_crop+80:end-h_crop, w_crop:end-w_crop);
```

Fig 3. Reajuste de la imagen y recorte

En este punto se etiquetaron los objetos de la imagen y se ordenaron de mayor a menor en función del área, obteniendo en primera posición la lente del microscopio que engloba a los gusanos y no es un objeto de interés, se realiza una máscara para este objeto y se elimina de la imagen, obteniendo el siguiente resultado.

```
% Aislar gusanos mediante mascaras

labels = bwlabel(clean);
prop = regionprops(labels, 'Area');

areas = [prop.Area];
[~, index] = sort(areas, 'descend');
edge_index = index(1);
mask_lente = ismember(labels, edge_index);
mask_cucs = clean & ~mask_lente;
```

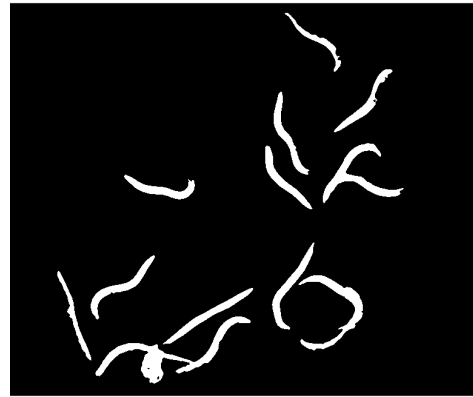


Fig 4. Aislamiento de gusanos mediante máscaras y resultado

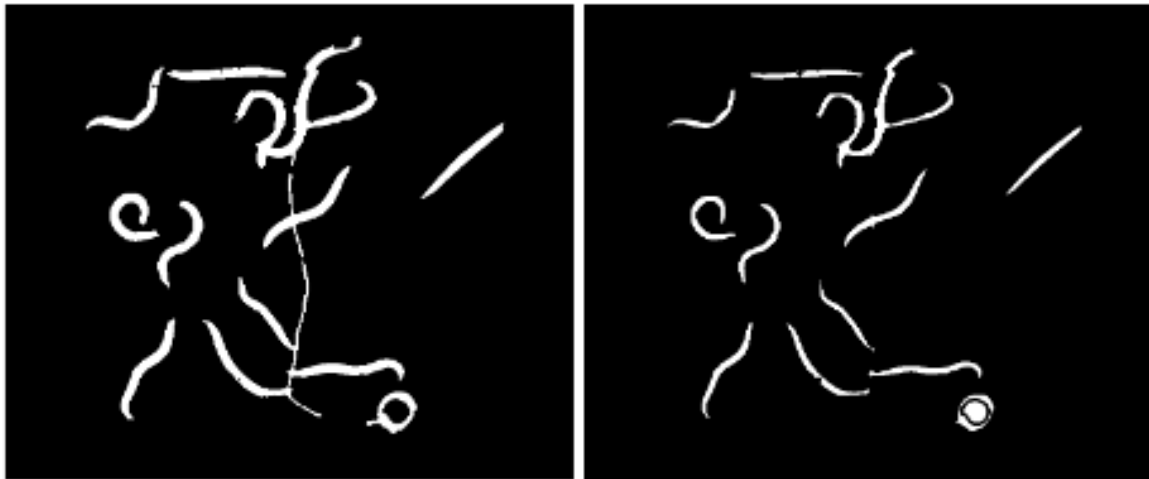
Una vez aislados los gusanos, se observa que algunos individuos están fusionados por lo que si se analiza la imagen se obtendrá un recuento incorrecto. Se trataron las imágenes con diversas operaciones morfológicas como erode o dilate pero los resultados obtenidos no fueron resolutivos en la mayoría de casos.

Para poder separar a los gusanos se planteó la siguiente estrategia. Se extrajeron los bordes de los gusanos y se realizó una operación de dilatación sobre ellos para hacerlos más gruesos, luego se rellenaron los gusanos y finalmente se restó la matriz de píxeles entre la objetos rellenados y los bordes dilatados, obteniendo unos gusanos mucho más finos que aún conservan su forma original. Surgieron inconvenientes ya que soluciona algunos casos pero siguen habiendo gusanos superpuestos que no se diferencian. Además, en gusanos vivos que están enroscados sobre sí mismos se crea un objeto nuevo circular en el interior del gusano, generando un error en el recuento.

```
%Deteccion de bordes y relleno
edges = edge(mask_cucs, 'Canny');
se = strel('disk', 2);
edges_open = imdilate(edges, se);
edges_filled = imfill(edges_open, 'holes');
pre_worms = edges_filled - edges_open;
worms = bwareaopen(pre_worms, 200);
```



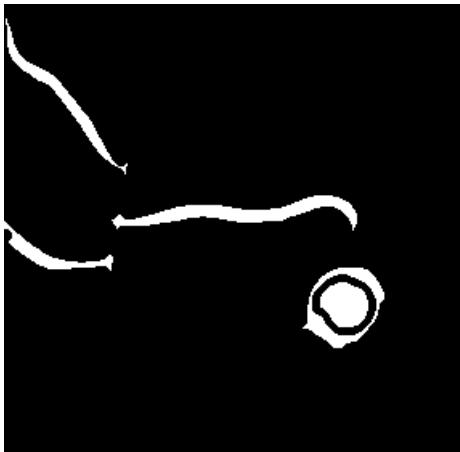
Fig 5. Código detección de bordes y resultado procesamiento con bordes



Img. 2.2 Comparación Imagen3 preprocesada vs Imagen3 después estrategia

Para arreglar este inconveniente generado se eliminarán los objetos circulares que se forman dentro de los gusanos si estos están enroscados. Estos casos suceden en las imágenes 3, 8, 9 y 10. Para eliminarlos se calcula la circularidad para cada objeto utilizando el área y el perímetro. Luego se etiquetan los objetos y se comprueba si la circularidad de algún objeto es cercana a 1, si es así el objeto se oculta.

```
%Eliminar posibles círculos generados
[worms_label, num_worms] = bwlabel(worms);
worms_props = regionprops(worms_label, 'Area', 'Perimeter');
tabla = struct2table(worms_props);
% Eliminar objetos con circularidad cercana a 1
for i = 1:num_worms
    circularity = 4*pi*worms_props(i).Area/worms_props(i).Perimeter^2;
    if abs(circularity - 1) < 0.3
        worms(worms_label == i) = 0;
    end
end
```



Una vez aplicada la estrategia se procede a clasificar las imágenes según la forma de los gusanos, ya que los gusanos vivos tienen una forma curvilínea mientras que los gusanos muertos tienen una forma rectilínea.

Para lograr esto se etiquetan los objetos de la imagen y se utiliza la relación entre la longitud del eje mayor y el eje menor de la elipse de cada objeto detectado. Ya que los gusanos muertos son rectilíneos, y con el tratamiento de bordes también muy finos, la relación entre su eje mayor y su eje menor será mucho mayor que la relación de los gusanos vivos, por lo que se establece un umbral de 10 para la clasificación. Si la relación es menor al umbral se considera que el objeto es curvilíneo y por tanto probablemente está vivo. Si la relación es mayor será rectilíneo y probablemente está muerto.

```
%CLASIFICAR SEGÚN LA FORMA
[cucs_label, num_cucs] = bwlabel(worms);
cucs_props = regionprops(cucs_label, 'Area','MajorAxisLength','MinorAxisLength')
tabla = struct2table(cucs_props);
relacion = tabla{:, 'MajorAxisLength'} ./ tabla{:, 'MinorAxisLength'};

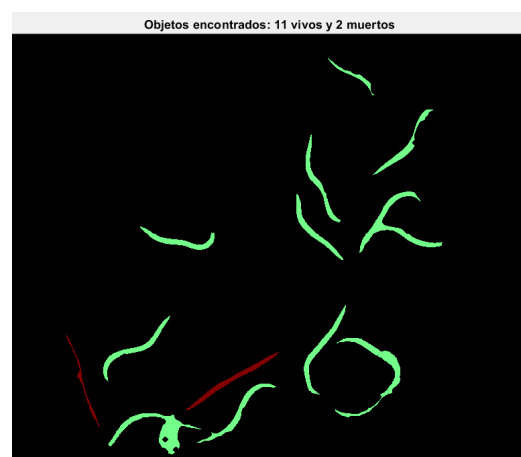
num_vivos = 0;
num_muertos = 0;

for i = 1:num_cucs

    if (cucs_props(i).MajorAxisLength/cucs_props(i).MinorAxisLength < 10)
        % Objeto curvilíneo (probablemente vivo)
        num_vivos = num_vivos + 1;
        cucs_label(cucs_label == i) = 125;
    else
        % Objeto rectilíneo (probablemente muerto)
        num_muertos = num_muertos + 1;
        cucs_label(cucs_label == i) = 255;
    end
end
```

Aplicando este procesamiento se obtiene una precisión del 100% en la clasificación de imágenes cuando se realiza la comprobación por lo que la clasificación funciona correctamente a nivel colectivo. Aún así el recuento de gusanos de manera individual no es correcto, por lo que se intentará mejorar la precisión en este campo.

MajorAxisLength	MinorAxisLength	Relacion
156.68	7.5267	20.816
147.91	24.217	6.1078
125.44	61.63	2.0354
129.76	22.627	5.735
169.6	9.455	17.938
148.29	23.129	6.4112
125.88	21.756	5.7859
153.52	28.94	5.3048
149.75	20.253	7.3937
87.488	10.796	8.1036
143.89	105.27	1.3668
122.98	102.94	1.1947
137.95	21.96	6.282



Para obtener una mejora notable, es necesario separar los gusanos que están superpuestos, ya que se cuentan como un solo conjunto en lugar de como individuos diferentes.

La estrategia a seguir para separar los gusanos que están unidos es salpicar operaciones morfológicas. Ya que se quieren separar los objetos será conveniente aplicar una erosión, el inconveniente de aplicarla se debe a que algunos de los gusanos que ya están aislados son muy finos y al aplicar la operación se eliminan por completo de la imagen.

```
% Procesar posibles gusanos unidos
for i = 1:num_gusanos
    if gusanos_props(i).Area > 1700
        % Erosion para separar los objetos
        se = strel('disk', 3);
        gusano = gusanos_label == i;
        gusano = imerode(gusano, se);
        joined_gusanos = joined_gusanos | gusano;
    elseif gusanos_props(i).Area > 1200
        se = strel('square', 2);
        gusano = gusanos_label == i;
        gusano = imerode(gusano, se);
        joined_gusanos = joined_gusanos | gusano;
    end
end

%Añadir gusanos pequeños
for i = 1:num_gusanos
    if gusanos_props(i).Area < 1200
        gusano = gusanos_label == i;
        small_gusanos = small_gusanos | gusano;
    end
end

% Unir las dos máscaras
gusanos = joined_gusanos | small_gusanos;
gusanos = bwareaopen(gusanos, 100);
```



Figura 8. Realización operaciones morfológicas en gusanos superpuestos

Para solucionarlo se recurre a las propiedades de los objetos, en este caso el área. Se crean dos máscaras vacías, en una se añadirán los gusanos con un área menor a un determinado lindar (1200). Se considera que gusanos menores a este área están aislados y diferenciados correctamente. Respecto a los gusanos con un área mayor al lindar, se les realiza una erosión para tratar de separar los gusanos, si es mayor a 1200 se aplica una erosión (square, 2) si el área es mayor a 1700 se realiza esta erosión y otra adicional (disk, 3) para intentar delimitar los gusanos superpuestos. Una vez separados se añaden a la otra máscara vacía y finalmente las dos máscaras se unen y se realiza la clasificación.

Area	MajorAxisLength	MinorAxisLength	Relacion
397	156.68	7.5267	20.816
1008	147.91	24.217	6.1078
129	67.912	11.223	6.0509
918	129.76	22.627	5.735
764	51.237	23.598	2.1712
1006	169.6	9.455	17.938
1002	148.29	23.129	6.4112
871	125.88	21.756	5.7859
1072	153.52	28.94	5.3048
913	149.75	20.253	7.3937
347	87.488	10.796	8.1036
145	61.89	10.239	6.0445
1087	121.43	46.73	2.5985
395	110.71	21.721	5.0971
829	137.95	21.96	6.282
163	76.119	9.5567	7.965

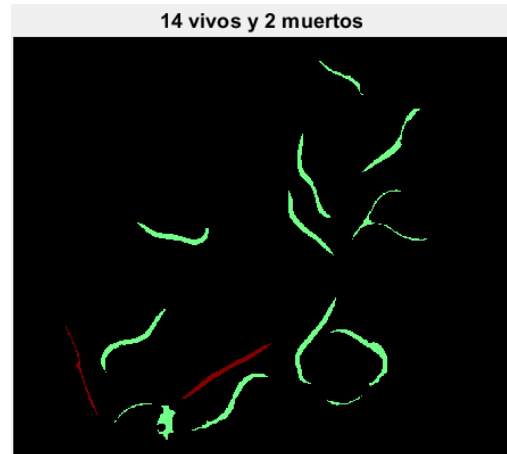


Figura 9. Tabla de propiedades de gusanos y resultado final

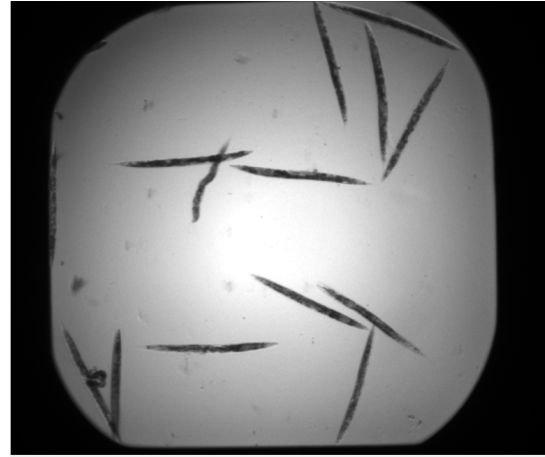
Finalmente después de todos los cambios en el preprocesamiento y la segmentación de la imagen, el esquema de la práctica sería el siguiente:

1. Preprocesamiento
 - a. Redimensionar y recortar
 - b. Aumentar Contraste
 - c. Filtro de Mediana para reducir ruido
 - d. Binarización local de la imagen
 - e. Limpieza de objetos con área pequeña
 - f. Limpieza del área de la lente
2. Detección de Bordes
 - a. Detección con operador Canny
 - b. Dilatación de bordes
 - c. Rellenado
 - d. Diferencia relleno y dilatación de bordes
3. Tratar gusanos unidos
 - a. Obtener propiedades de los objetos
 - b. Tratar gusanos más grandes
 - c. Unificar gusanos pequeños y grandes divididos
4. Clasificar según la forma
 - a. Obtener propiedades de los gusanos
 - b. Clasificar según la relación entre su eje mayor y eje menor

3. Evolución de una de las imágenes durante el preprocesamiento

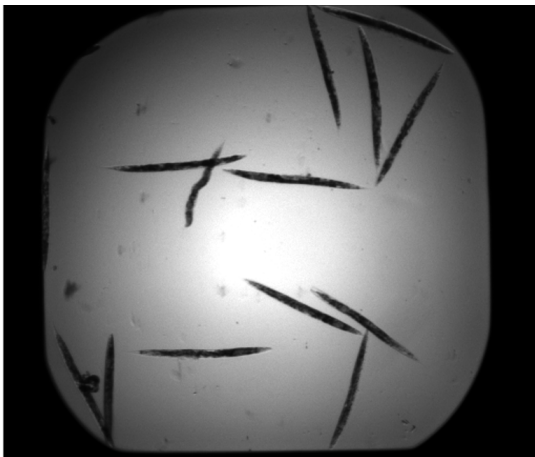


Img 3.1 Imagen Original



Img 3.2 Imagen ajustada y recortada

Se realiza un reescalado de la imagen y se recortan los bordes para obtener una imagen donde se ha eliminado gran parte del background que no es interesante para la clasificación y donde los gusanos se ven mejor definidos.

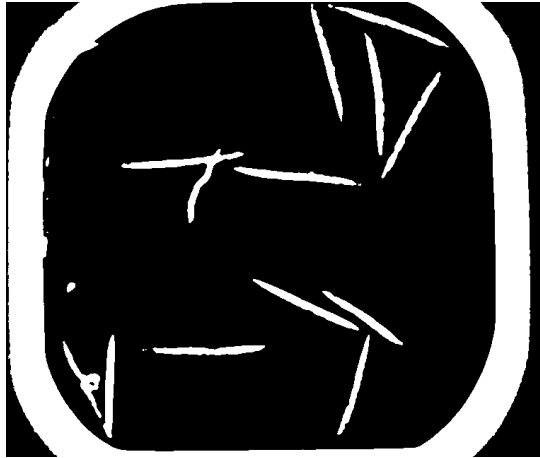


Img 3.3 Contraste



Img 3.4 Filtro suavizado de mediana

Se realiza un aumento del contraste de la imagen para resaltar los píxeles más oscuros de la imagen y se reduce el ruido con un filtro de mediana. se binariza la imagen mediante una umbralización adaptativa invertida, por lo que se obtendrán objetos blancos sobre un fondo negro. Luego se eliminan los objetos pequeños que no pertenecen a los gusanos.

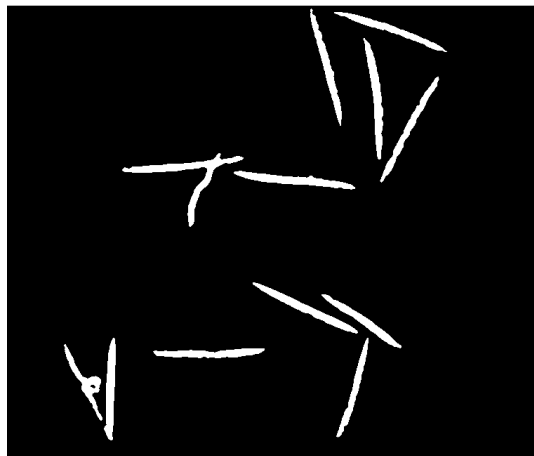


Img 3.5 Binarización local



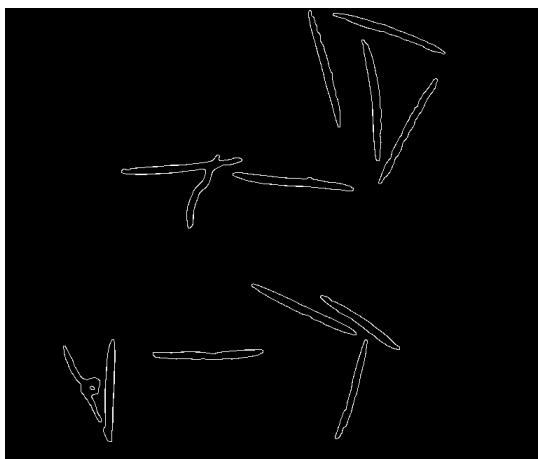
Img 3.6 Limpiar pequeñas áreas

Se binariza la imagen mediante una umbralización adaptativa invertida, por lo que se obtendrán objetos blancos sobre un fondo negro. Luego se eliminan los objetos pequeños con un área menor a 200, restos de la imagen que no pertenecen a los gusanos.

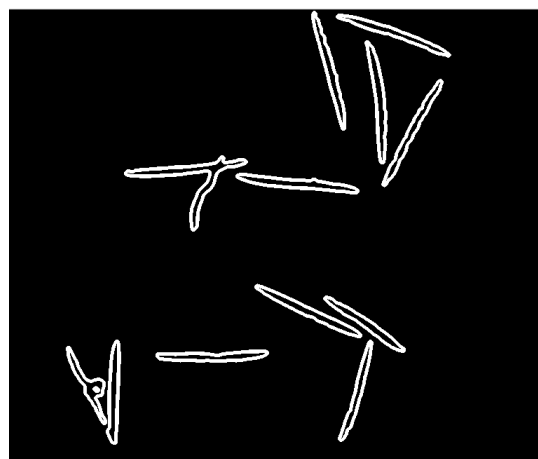


Img 3.7 Gusanos aislados

Eliminar la máscara equivalente a la lente del microscopio para aislar los gusanos
Se realiza una máscara de la lente del microscopio, equivalente al objeto de mayor área de la imagen, luego se elimina esta máscara de la imagen binaria para aislar los gusanos.

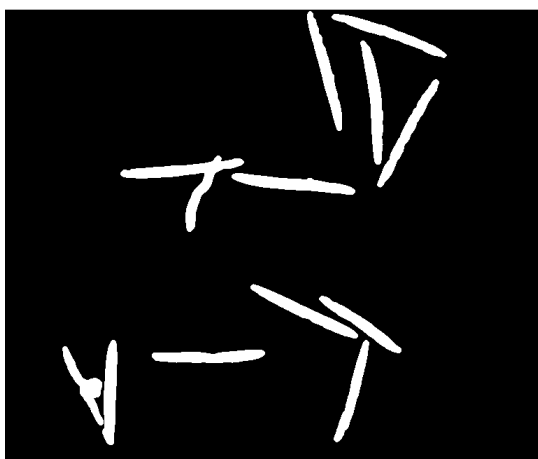


Img 3.8 Detección de contornos



Img 3.9 Dilatación de los bordes

Primero se utiliza la función edge para detectar los bordes de los objetos en la imagen. Se realiza una dilatación de los bordes con un elemento estructurado disk.



Img 3.10 Gusanos rellenados



Img 3.11 Gusanos aislados y finos

Se realiza una operación de relleno y finalmente se resta a estos objetos rellenados el borde dilatado. Se obtienen unos gusanos que mantienen la forma, pero son más finos, permitiendo separar algunos gusanos que se tocan y detectandolos como individuos distintos.

El preproceso realizado hasta esta etapa se transforma en la primera versión del juego de pruebas ya que la precisión en la clasificación es del 100%.

Para esta imagen no se detectan formas circulares y pese a que se observan dos gusanos que colisionan en algún punto el área no es suficientemente grande como para tratarlos sin erosionar alguno de los gusanos. El resto del procedimiento está detallado en el apartado de "Diseño e Implementación".

4. Juego de Pruebas

Para realizar el juego de pruebas se realizó el conteo de gusanos vivos/muertos de forma manual y se ejecutaron las tres versiones del código.

V1: Preprocesamiento hasta detección de bordes y clasificación según proporción de la longitud entre el eje mayor y el eje menor.

V2: Se tratan los gusanos con un área considerablemente mayor, se obtienen los mejores resultados.

V3: Se eliminan los círculos formados que se contabilizan como gusanos, esto disminuye la cantidad de gusanos pero aumenta la fiabilidad ya que se asegura detectar solo los gusanos.

	<i>V teóricos</i>	<i>M teóricos</i>	<i>Vv1</i>	<i>Mv1</i>	<i>Vv2</i>	<i>Mv2</i>	<i>Vv3</i>	<i>Mv3</i>
1	12	4	9	4	9	5	9	5
2	12	4	12	1	12	1	12	1
3	14	1	9	4	10	4	9	4
4	12	2	7	3	6	5	6	5
5	11	3	9	3	10	3	9	3
6	10	5	9	3	11	3	11	3
7	11	3	7	2	8	4	8	4
8	12	3	10	1	12	2	12	4
9	11	4	10	3	12	4	12	4
10	9	3	7	6	9	5	8	5
11	14	1	11	2	14	2	14	2
12	11	2	9	2	9	3	9	3
13	4	9	4	5	4	5	4	5
14	2	12	2	8	2	8	2	8
15	0	14	2	11	2	11	2	11
16	1	11	5	6	5	6	5	6
17	2	11	2	10	2	10	2	10
18	0	13	1	14	1	14	1	14
19	0	12	4	5	5	6	5	6
20	1	13	2	7	2	9	2	9
21	0	13	3	5	3	7	3	7
22	0	13	1	6	1	9	1	9
23	0	12	1	8	1	8	1	8
24	1	13	2	11	2	11	2	11

5. BIBLIOGRAFÍA

<https://es.mathworks.com/>

<https://chat.openai.com/>