



UNIVERSIDAD POLITÉCNICA SALESIANA

SEDE GUAYAQUIL

CARRERA DE INGENIERÍA EN ELECTRONICA Y AUTOMATIZACIÓN

**IMPLEMENTACIÓN DE PRÁCTICAS MEDIANTE SOFTWARE DE PROCESAMIENTO
DE IMÁGENES PARA LA DETECCIÓN, MEDIDA DE MICRO OBJETOS Y
COMPARACIÓN DE PUNTOS DE INTERÉS CON UN MICROSCOPIO DIGITAL**

Trabajo de titulación previo a la obtención del
título de Ingeniero en Electrónica y Automatización

AUTORES: CARLOS JOSÉ NARANJO MORÁN

ISAAC MIGUEL OCHOA BERNABÉ

TUTOR: ING. BREMNEN MARINO VELIZ NOBOA PHD.

Guayaquil – Ecuador

2023

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN

Nosotros, Carlos José Naranjo Morán con documento de identificación N° 0950941138 e Isaac Miguel Ochoa Bernabé con documento de identificación N° 0954605630; manifestamos que:

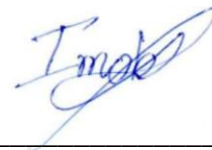
Somos los autores y responsables del presente trabajo; y, autorizo a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación.

Guayaquil, 2 de marzo del 2023

Atentamente,



Carlos José Naranjo Morán
0950941138



Isaac Miguel Ochoa Bernabé
0954605630

CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA

Nosotros, Carlos José Naranjo Morán con documento de identificación N° 0950941138 e Isaac Miguel Ochoa Bernabé con documento de identificación N° 0954605630, expresamos nuestra voluntad y por medio del presente documento cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del Proyecto técnico: “implementación de prácticas mediante software de procesamiento de imágenes para la detección, medida de micro objetos y comparación de puntos de interés con un microscopio digital”, el cual ha sido desarrollado para optar por el título de: Ingeniero de Sistemas, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribimos este documento en el momento que hacemos la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

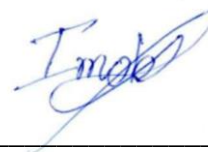
Guayaquil, 2 de marzo del 2023

Atentamente,



Carlos José Naranjo Morán

0950941138



Isaac Miguel Ochoa Bernabé

0954605630

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, Bremnen Marino Veliz Noboa con documento de identificación N° 0703865139, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: IMPLEMENTACIÓN DE PRÁCTICAS MEDIANTE SOFTWARE DE PROCESAMIENTO DE IMÁGENES PARA LA DETECCIÓN, MEDIDA DE MICRO OBJETOS Y COMPARACIÓN DE PUNTOS DE INTERÉS CON UN MICROSCOPIO DIGITAL, realizado por Carlos José Naranjo Morán con documento de identificación N° 0950941138 y por Isaac Miguel Ochoa Bernabé con documento de identificación N° 0954605630, obteniendo como resultado final el trabajo de titulación bajo la opción Proyecto técnico que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Guayaquil, 2 de marzo del 2023

Atentamente,



Ing. Bremnen Marino Veliz Noboa PHD

0703865139

DEDICATORIA

Dedico este trabajo a mi familia por su apoyo a lo largo de mi carrera universitaria y durante toda mi vida. Gracias a todas las personas especiales que han estado conmigo durante esta etapa y que han contribuido a mi formación profesional y humana. Agradezco a Dios por ser mi guía en mi vida, acompañándome, dándome paciencia y sabiduría, para que pueda culminar con éxito las metas que me propongo. Todo este esfuerzo se lo dedico a mi querida madre, porque sé que ella siempre me ha sacado adelante y siempre me ha ayudado, además de darme la vida, siempre ha creído en mí y nunca me ha abandonado. Mamá te amo.

Carlos José Naranjo Morán

Este trabajo está dedicado principalmente a Dios quien supo guiarme por este largo camino de formación profesional, por darme fuerzas y no permitirme desmayar ante las adversidades. A mis padres ya que han sido mi reflejo de esfuerzo diario y ejemplo de perseverancia, son ellos mi motor para cumplir mis metas, con acertados consejos, valores y principios me han dado impulso para salir adelante permitiéndome alcanzar un triunfo más. A mi abuela Rosita quien con su sabiduría, afecto y apoyo incondicional me motiva a seguir luchando hacia un futuro mejor. A mis hermanos y a esas personas que estuvieron apoyándome en cada decisión, que tuvieron paciencia y entrega hacia mí, a esas personas les dedico y agradezco, porque hoy con alegría presento y disfruto este trabajo y el más anhelado logro.

Isaac Miguel Ochoa Bernabé

AGRADECIMIENTO

Un agradecimiento especial a la Universidad Politécnica Salesiana y a sus distinguidos docentes que en aula me brindaron sus conocimientos y enseñanzas. Compartiendo cada una de sus experiencias, que nos serán útil para evitar que el mundo laboral nos engulle. Agradecerles por toda la paciencia, tolerancia y buenos consejos que nos brindaron a lo largo de esta carrera y a cada una de las personas que siempre estuvieron con nosotros durante el recorrido de este largo camino para poder cumplir esta meta en común.

Carlos José Naranjo Morán

Isaac Miguel Ochoa Bernabé

Índice

1. Introducción	1
1.1 Problema	1
2. Objetivos	3
2.1 Objetivo general	3
2.2 Objetivo específico	3
3. Marco teórico	4
4. Metodología	9
4.1 Microscopio digital	9
4.1.1 ANNLOV Microscopio digital.....	9
4.2 Binarización de una imagen.	11
4.3 Comparación de puntos interés.....	12
4.4 Detección y medición de micro objetos.	13
5. Resultados	14
Practica #1	14
Tema.....	14
Objetivos	14
Materiales y equipo	14
Desarrollo	15
Recomendaciones.....	21
Practica #2	22

Tema.....	22
Objetivos	22
Materiales y equipo	22
Desarrollo	23
Recomendaciones.....	26
PRACTICA #3	27
Tema.....	27
Objetivos	27
Materiales y equipo	27
Desarrollo	28
Recomendaciones.....	33
6. Conclusión.....	34
7. Cronograma y actividades para desarrollar	35
8. Presupuesto.....	35
9. Bibliografía.....	35

1. Introducción

La implementación de estas prácticas, a través de la visión artificial nos ayudara a obtener resultados precisos de los diferentes tipos de procesamiento de imágenes. Obteniendo por medio del software detalles de los pixeles en la imagen binarizadas, comparación de puntos de interés y detección - medición de micro objetos.

1.1 Problema

La visión humana no es capaz de detectar, analizar y almacenar imágenes con un gran poder de procesamiento como se puede realizar con un sistema digital. En la asignatura de procesamiento digital de señales hay un apartado corto sobre procesamiento de imágenes, pero los estudiantes no tienen muchas practicas relacionadas a esta temática. Por otro lado, la electrónica de hoy día cada vez es de componentes más pequeños que requieren por ejemplo de microscopio para una buena visión.

La presente propuesta aporta con la elaboración de un banco de pruebas mediante códigos en Matlab que enriquecen el contenido de la asignatura. Se realizarán prácticas de diferentes métodos de procesamiento de imágenes empleando la técnica de visión artificial para micro objetos a través de un microscopio digital.

Este proyecto va a tener muchos beneficios al aprendizaje de lo que es el procesamiento de imágenes en la asignatura de procesamiento de señales digitales. Una de las practicas será detectar y medir micro objetos que son muy complicados de observar y medir manualmente. El microscopio digital será donado al laboratorio de Física y Nanotecnología que tiene la Universidad Politécnica Salesiana.

Con el microscopio óptico que hemos hablado, podremos extraer información necesaria de una imagen digital. Actualmente, existen distintos sistemas comerciales, o cámaras

inteligentes, que tienen herramientas para realizar el procesado de las imágenes. Muchas veces estas herramientas son suficientes para una aplicación concreta, sin embargo, otras veces, es necesario desarrollar un software específico que aporte la versatilidad necesaria.

Tras el procesado de las imágenes y una vez se ha obtenido el resultado o la Información necesaria, esta información es devuelta al exterior como output del sistema.

2. Objetivos

2.1 Objetivo general

- Implementar prácticas mediante software de procesamiento imágenes para la detección, medida de micro objetos y comparación de puntos de interés con un microscopio digital.

2.2 Objetivo específico

- Programar con el software Matlab códigos para captar e interpretar imágenes.
- Detectar y medir micro objetos de una imagen tomada con el microscopio digital.
- Comparar puntos de interés de una imagen tomada con el microscopio digital.
- Realizar 3 prácticas para estudiantes con los códigos obtenidos.

3. Marco teórico

La binarización es el proceso de reducir la información a un único valor, donde todos los valores posibles son verdadero y falso (1 y 0). En el caso de una imagen digital, verdadero y falso corresponden a dos colores: blanco y negro. [1]

La binarización de una imagen suele hacerse utilizando un umbral en la escala de grises para determinar si un píxel es blanco o negro. Por lo general, las imágenes no están en una escala de grises, por lo que tendrá que estimar el color de los grises calculando el brillo medio en toda la imagen. También puede especificar umbrales basados en otros tonos, por ejemplo, podemos calcular qué color será después de mezclar dos colores específicos.

Una imagen digital se considera como una cuadrícula. Cada elemento de esa cuadrícula se llama Píxel (Picture element). [2] Las imágenes digitales, al igual que el resto de la información digital, se representan mediante valores numéricos. Consiste en dividir una imagen en trozos rectangulares que son los denominados píxeles. A cada píxel se le asigna un número y se determina el valor medio de todos los píxeles que lo rodean.

La conectividad entre píxeles es un concepto para establecer los límites en objetos y regiones de componentes en una imagen. Para establecer la conectividad entre dos píxeles, es necesario determinar si son adyacentes en sentido específico (en el caso que tenga 4 vecindades) y si su nivel de escala de gris satisface un criterio especificado (si son iguales). Por ejemplo, una imagen binarizada con valores de 0 y 1, dos píxeles pueden tener una vecindad de 4.

Cuando hablamos de conectividad-4, sólo están conectados los píxeles que son N, W, E, S del píxel dado. Sin embargo, si además hay que considerar los píxeles de las diagonales, entonces tenemos conectividad- 8 (es decir, N, NW, W, NE, SE y E están todos conectados).

En la **Figura 1** (derecha), utilizamos este concepto para determinar si la región A y la región B están conectadas y utilizamos un modelo de conectividad local (aquí $N \times N=3 \times 3$) para determinar si se trata de características de imagen separadas o de la misma característica. Las operaciones realizadas localmente en las imágenes, como el filtrado y la detección de bordes, tienen en cuenta la ubicación de un píxel dado (i,j) en términos de su vecindad local de píxeles indexada como un desplazamiento $(i \pm k; j \pm k)$. El tamaño y la escala de estos vecindarios pueden controlarse variando el parámetro N del tamaño del vecindario a partir del cual se calcula un desplazamiento k (generalmente $[N=2]$).

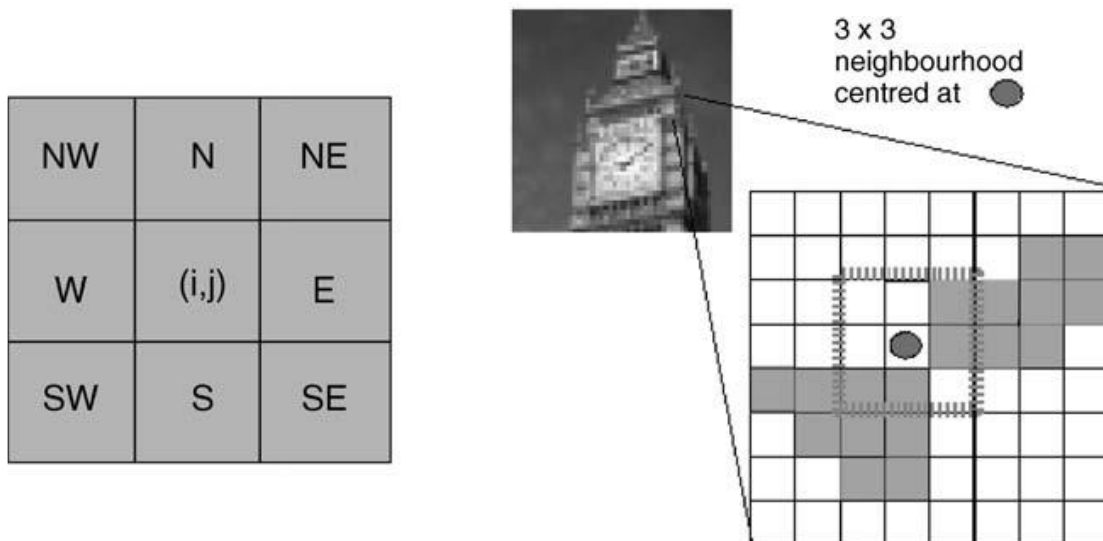


Figura 1. La imagen superior izquierda muestra la conectividad de un vecindario 3×3 centrado en un píxel específico. La imagen inferior derecha muestra un ejemplo de una vecindad 3×3 centrada en otra ubicación de píxel específica.

[3]

En el procesamiento de imágenes, la conectividad 8 es el esquema de conectividad más utilizado. También es el predeterminado en la mayoría de los procesadores modernos, y para un tamaño razonable de vecindad, suele ser alcanzable en tiempo real en los procesadores modernos para la mayoría de las operaciones.

Las operaciones de filtrado sobre una imagen completa se realizan generalmente como una serie de operaciones de vecindad local utilizando un principio basado en ventanas deslizantes: todos y cada uno de los píxeles de la imagen se procesan basándose en una operación realizada en su vecindad local de $N \times N$ píxeles (región de influencia).

El concepto de momentos forma una parte importante de la teoría elemental de la probabilidad. Si nosotros tenemos una función de densidad de probabilidad ($p(x)$) que describe la distribución de una variable x , entonces el “ n th” momento se define como

$$m_n = \int_{-\infty}^{\infty} x^n p(x) dx \quad [3]$$

El momento cero $m_0 = \int_{-\infty}^{\infty} x^0 p(x) dx$ da el área total bajo la función ($p(x)$), y es siempre igual a la unidad si ($p(x)$) es una verdadera función de densidad de probabilidad. El primer momento, $\mu = m_1 = \int_{-\infty}^{\infty} x^1 p(x) dx$, corresponde al valor medio de la variable aleatoria.

Los momentos centrales de la función de densidad describen la variación con respecto a la media y se definen como:

$$M_n = \int_{-\infty}^{\infty} (x - \mu)^n p(x) dx \quad [3]$$

El momento central más común, $M_2 = \int_{-\infty}^{\infty} (x-\mu)^2 p(x) dx$, es lo conocido como la varianza y forma la medida más básica de cuán "dispersa" es la función de densidad.

Los momentos de orden superior pueden proporcionar otra información sobre la forma de la función de densidad, como la asimetría (la tendencia a disparar más en un lado de la media que en el otro).

Un teorema importante de la teoría de la probabilidad establece que el conocimiento de todos los momentos determina de forma única la función de densidad. Así, podemos entender que los momentos codifican colectivamente información sobre la forma de la función de densidad.

Los momentos se extienden naturalmente a las funciones 2-D (y dimensiones superiores). Por lo tanto, el pq th momento de una función de densidad 2-D $p(x, y)$ está dada por

$$M_{n= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q p(x, y) dx dy} \quad [3]$$

Dado que los momentos centrales se miden con respecto al centroide de la forma, son necesariamente invariantes con la traslación.

Sin embargo, necesitamos descriptores que no cambien cuando la forma se escala y/o rota, es decir, invariantes de escala y rotación. Está fuera del alcance de nuestra discusión inmediata ofrecer una prueba, pero es posible demostrar que los momentos centrales normalizados poseen invariancia de escala.

El momento central normalizado $(p - q)$ th se define como:

$$n_{pq} = \frac{M_{pq}}{M_{00}^\beta} \text{ donde } \beta = \frac{p+q}{2} + 1 \wedge p+q \geq 2$$

[3]

A partir de estos momentos centrales normalizados, es posible calcular siete magnitudes derivadas atribuidas a Hu (también denominadas momentos) que son invariantes a la traslación, la escala y la rotación. Con ellas se puede determinar la posición de equilibrio del objeto en cuestión:

$$\begin{aligned} \Lambda_1 &= \eta_{20} + \eta_{02} \\ \Lambda_2 &= (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \\ \Lambda_3 &= (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \\ \Lambda_4 &= (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{30})^2 \\ \Lambda_5 &= (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} - \eta_{03})^2] + (3\eta_{21} - \eta_{03})(\eta_{03} + \eta_{21}) \\ &\quad \times [3(\eta_{30} + \eta_{12})^2 - (\eta_{03} + \eta_{21})^2] \\ \Lambda_6 &= (\eta_{20} - \eta_{02})[(\eta_{12} + \eta_{30})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{21} + \eta_{03})(\eta_{12} + \eta_{30}) \\ \Lambda_7 &= (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{03} + \eta_{21})^2] + (3\eta_{21} - \eta_{30})(\eta_{21} + \eta_{03}) \\ &\quad \times [3(\eta_{30} + \eta_{12})^2 - (\eta_{03} + \eta_{21})^2] \end{aligned}$$

[3]

La **Figura 2.** muestra un objeto extraído de una imagen y otro que ha sido escalado, rotado y trasladado con respecto al primero. Los valores de los tres primeros momentos se calculan según ecuaciones para las tres formas. Como es evidente, son iguales dentro de la precisión computacional que permite una implementación discreta digital de estas ecuaciones en la Ecuación. [3]

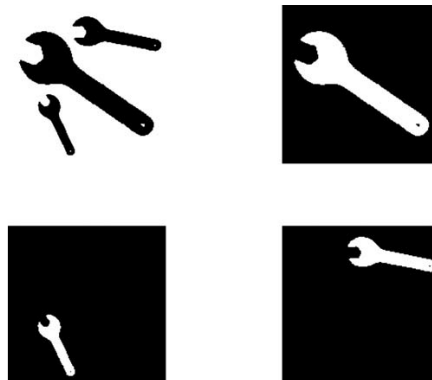


Figure 2. Objetos de ejemplo para el cálculo del momento Hu.

4. Metodología

4.1 Microscopio digital

Los microscopios digitales son microscopios que funcionan sin oculares. Una cámara digital actúa como detector. Las imágenes se visualizan en una pantalla o monitor, de forma que el puesto de trabajo del microscopio se convierte en un ergonómico puesto de ordenador. [4]

4.1.1 ANNLOV Microscopio digital

El microscopio digital LCD de 7 pulgadas de ANNLOV adopta una cámara de enfoque de 12MP y una pantalla de alta definición de 1080P, que puede producir imágenes y videos de alta calidad. El diseño de pantalla de 90 grados le permite encontrar el ángulo de visión más cómodo y elimina la tensión de los ojos y el cuello.

Permite acercar y ver detalles increíbles de 50x a 1200x. La ampliación real depende del tamaño de la pantalla, la distancia entre la cámara y los puntos de visualización.

8 LED regulables hacen que la mayoría de las partes sean brillantes y claras, pero algunas partes aún son muy oscuras e invisibles, 2 luces laterales regulables lo suficientemente flexibles y pueden iluminarse en diferentes ángulos, asegurando que cada detalle sea claro y nítido.

Captura de fotos y videos: el microscopio ANNLOV puede capturar imágenes de 12 MP (4023 x 3024), grabar videos de 1080 FHD 1920 x 1080 30 fps y guardar en una tarjeta microSD de 32 GB. [5]



Figura 3. Microscopio digital ANNLOV.

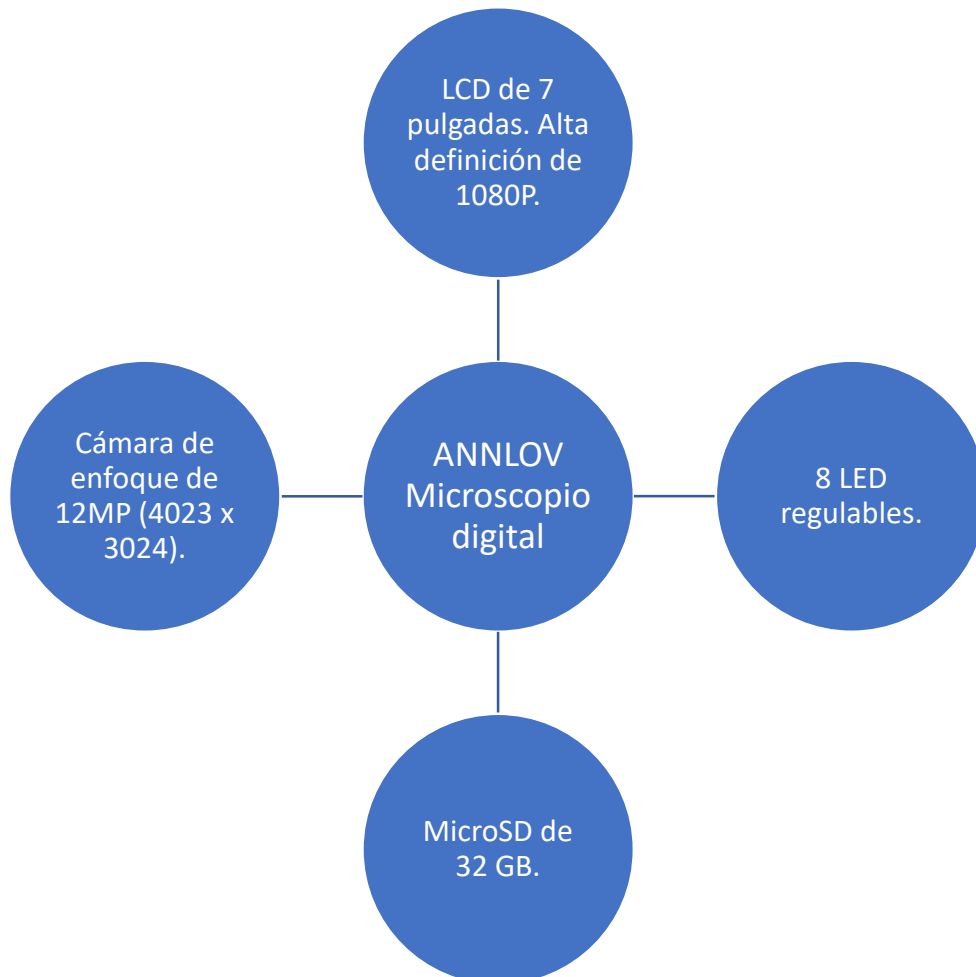


Figura 4. Diagrama microscopio digital ANNLOV.

4.2 Binarización de una imagen.

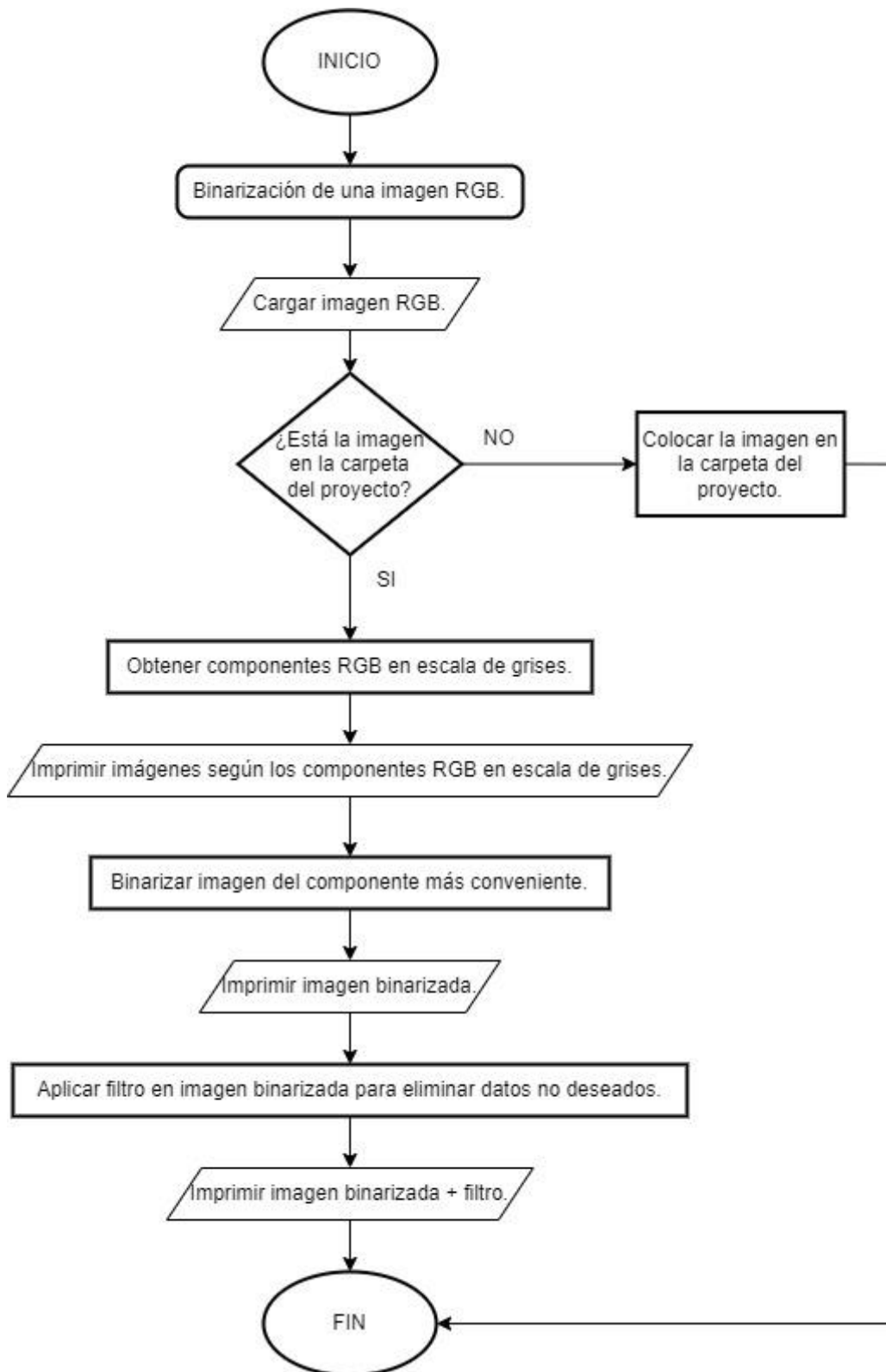


Figura 5. Diagrama de flujo practica #1.

4.3 Comparación de puntos interés.

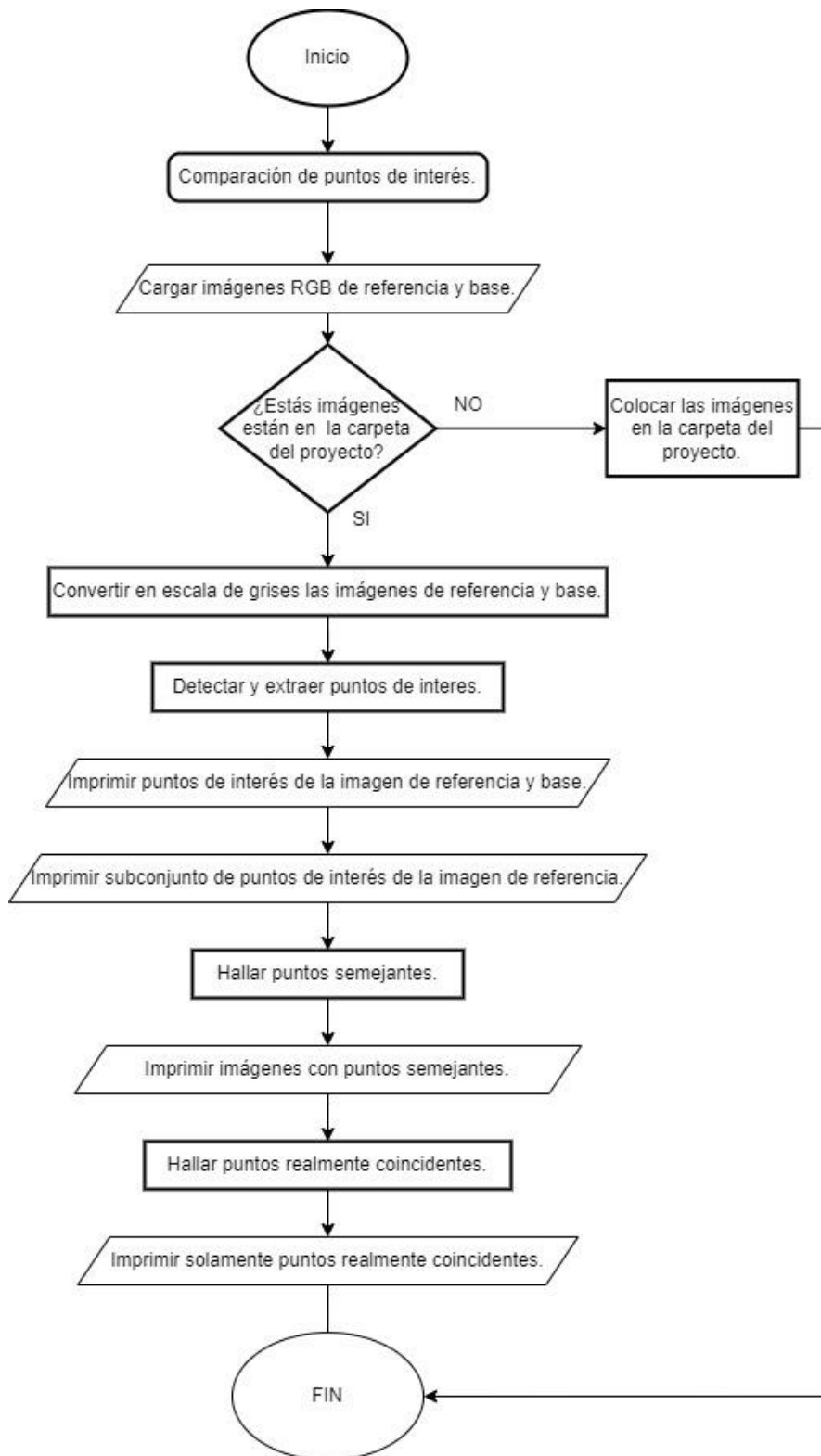


Figura 6. Diagrama de flujo practica #2.

4.4 Detección y medición de micro objetos.

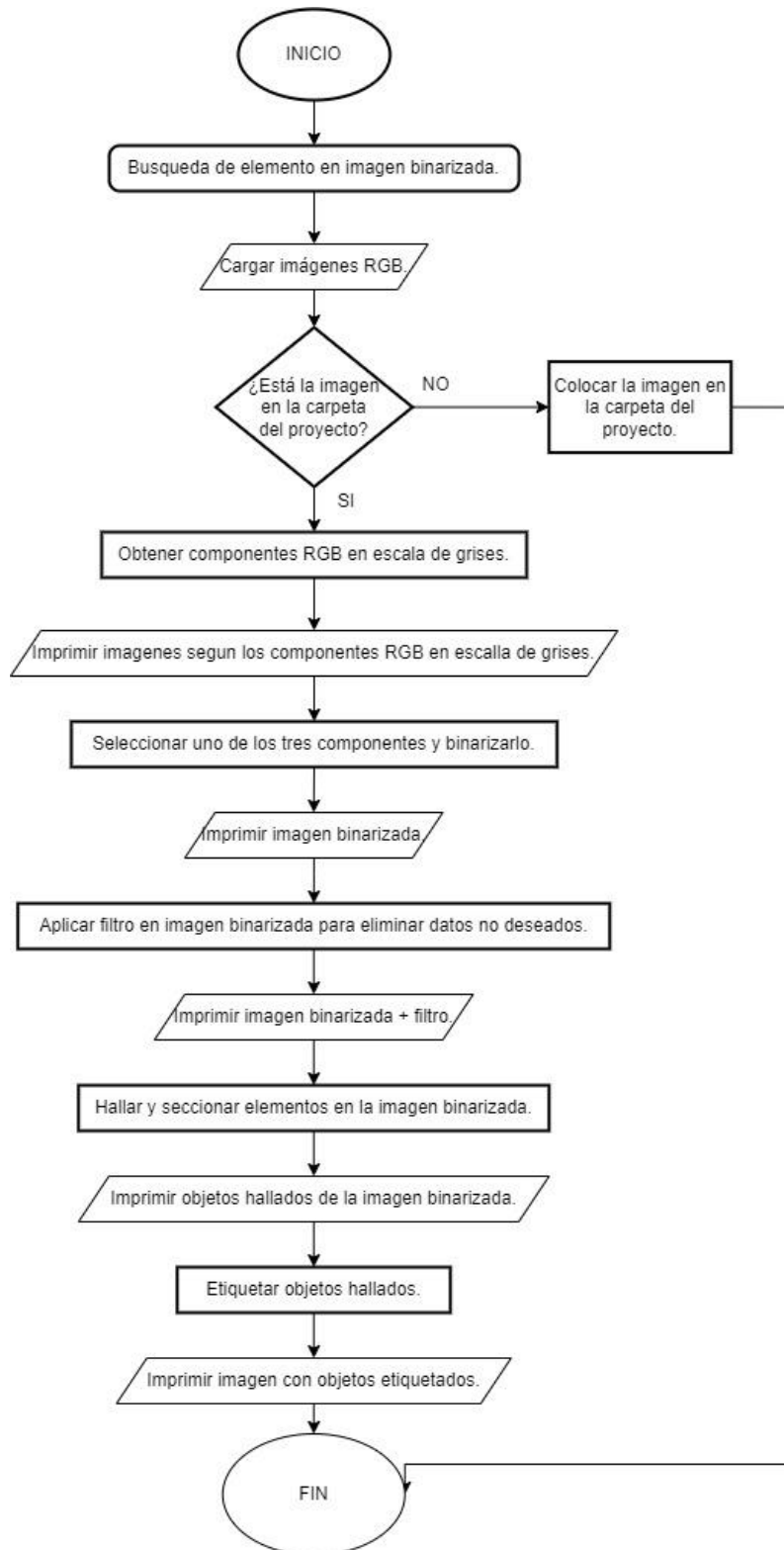


Figura 7. Diagrama de flujo practica #3.

5. Resultados

Practica #1

Tema

Manejo del microscopio y Binarización de una imagen.

Objetivos

Objetivo general

- Implementar prácticas mediante software de procesamiento imágenes para la detección, medida de micro objetos y comparación de puntos de interés con un microscopio digital.

Objetivo específico

- Programar con el software Matlab códigos para captar e interpretar imágenes.

Materiales y equipo

- Ordenador o laptop.
- Matlab.
- Microscopio digital ANNLOV.

Desarrollo

El microscopio digital cuenta con un mando alámbrico para comodidad del operador al ejecutar la captura de imágenes y videos. Esta cuenta de dos conexiones, una de ellas conectada al lateral de la pantalla, cuyo objetivo es el control principal del microscopio y la segunda conexión es a la base, que tiene como propósito colaborar con luz led controlada.



Figura 8. Microscopio digital ANNLOV.



Figura 9. Microscopio digital ANNLOV.

Cuenta con una memoria SD de 32GB, donde se guardaran nuestras imágenes y videos. La pantalla posee la ranura SD en la parte lateral de la misma. La transferencia de datos se hace a través de la Micro SD al ordenador o portátil.



Figura 10. Microscopio digital ANNLOV.



Figura 11. Microscopio digital ANNLOV.

El microscopio se enciende manteniendo presionado por unos segundos el pulsador de la esquina derecha. Este por default enciende en modo captura de imágenes. Pulsando “OK” o el símbolo de la Cámara en el mando, podremos capturar imágenes sin inconvenientes.



Figura 12. Microscopio digital ANNLOV.



Figura 13. Microscopio digital ANNLOV.

Binarización de una imagen. Cargamos e imprimimos la imagen RGB que se capturo en la **Figura 12**. Esta imagen previamente debe estar en la carpeta de nuestro proyecto. Caso contrario no se ejecutará con éxito.

Binarización de una imagen.

```
clear
close all
clc
I=imread('PICA0036.jpg'); % Carga la imagen RGB.
figure(1)
imshow(I); %Imprime imagen RGB.
```

Figura 14. Script 1.



Figura 15. Imagen RGB.

Extraemos de la imagen sus componentes rojo, verde y azul en escala de grises. Imprimimos por separado las muestras **Figura 17**. Permittiéndonos analizarlas y posteriormente seleccionar la más accesible a binarizar. El valor numerico de una imagen en escala de grises oscila entre el blanco (255) y el negro (0).

Binarización de una imagen.

```
R=I(:,:,1); %Toma la componente en rojo en escala de grises de la matriz RGB.
G=I(:,:,2); %Toma la componente en verde en escala de grises de la matriz RGB.
B=I(:,:,3); %Toma la componente en azul en escala de grises de la matriz RGB.
f2 = figure(2); set(f2, 'color', 'w');
subplot(1,3,1); % Selecciona el 1 subplot.
imshow(R); title('Componente Rojo en escala de grises'); % Visualiza componente R.
colorbar
subplot(1,3,2); % Selecciona el 2 subplot.
imshow(G); title('Componente Verde en escala de grises'); % Visualiza componente G.
colorbar
subplot(1,3,3); % Selecciona el 3 subplot.
imshow(B); title('Componente Azul en escala de grises'); % visualiza componente B.
colorbar
```

Figura 16. Script 2.



Figura 17. Descomposición de componentes RGB

El umbral de binarización es calculado por el método Otsu, el cual minimiza la varianza interclas de los píxeles blancos y negros pasados por el umbral [6]. Imprimimos en una sola imagen la etapa de binarización del componente en rojo **Figura 19**.

Binarización de una imagen.

```
%Calculo del umbral de binarización del componente en rojo en escala de grises.  
nivel=graythresh(R);  
bw=imbinarize(R,nivel); %Binarización del componente en rojo.  
figure(3)  
imshowpair(R,bw,'montage');  
%Imprime imagen del antes y después del componente en su etapa de binarización.
```

Figura 18. Script 3.

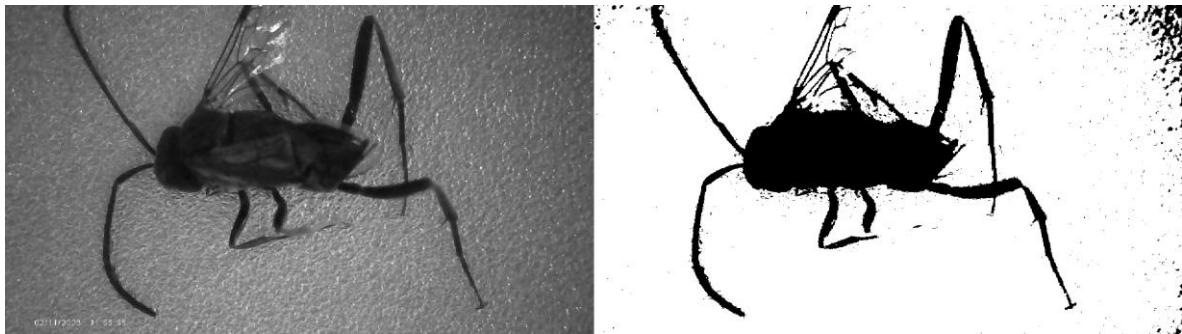


Figura 19. Imagen binarizada del componente en rojo.

Aplicamos el filtro a la imagen binarizada, posteriormente se imprime la imagen final. El filtro elimina objetos que contengan menos de 100 píxeles. Este argumento puede variar, según la calidad de píxeles que se desee en la imagen.

Binarización de una imagen

```
bwm1=bwareaopen(bw,100); % Aplicación del filtro.  
figure(4)  
imshow(bwm1); %Imprime imagen con el filtro aplicado.
```

Figura 20. Script 4.



Figura 21. Imagen binarizada + filtro.

Recomendaciones

Se recomienda tomar un tiempo prudente en la captura de imágenes con el microscopio digital. Capturar imágenes en diferentes ángulos e intensidades de brillo, antes de aplicar el procesamiento de imágenes.

Practica #2

Tema

Comparación de puntos de interés.

Objetivos

Objetivo general

- Implementar prácticas mediante software de procesamiento imágenes para la detección, medida de micro objetos y comparación de puntos de interés con un microscopio digital.

Objetivo específico

- Comparar puntos de interés de una imagen tomada con el microscopio digital.

Materiales y equipo

- Ordenador o laptop.
- Matlab.
- Microscopio digital ANNLOV.

Desarrollo

Comparación de puntos de interés. Cargamos e imprimimos las imágenes RGB. Estas imágenes vamos a categorizarlas como imagen de referencia y base. Ambas imágenes son capturadas de forma independiente. Los elementos capturados en la imagen de referencia son un subconjunto de elementos que se encuentran en la imagen base. Es decir, son imágenes totalmente independientes, pero una está dentro de otra. Previamente deben estar en la carpeta de nuestro proyecto, caso contrario no se ejecutará con éxito.

Comparación de puntos de interés.

```
clear;
clc;
close all;
ref_img = imread('microscopio.jpeg'); %Carga la imagen de referencia en RGB.
base_img = imread('microscopio_2.jpeg'); %Carga la imagen de base en RGB.
```

Figura 22. Script 5.

Procesamos a escala de grises las imágenes RGB previamente cargadas.

Comparación de puntos de interés

```
ref_img_gray = rgb2gray(ref_img); %Convierte a escala de grises la imagen de referencia.
base_img_gray = rgb2gray(base_img); %Convierte a escala de grises la imagen de base.
```

Figura 23. Script 6.

Extraemos e imprimimos puntos de interés en las imágenes de referencia **Figura 25** y base **Figura 26**. Utilizando el algoritmo de características robustas aceleradas (SURF) [7].

Comparación de puntos de interés.

```
%Imagen de referencia detección de características.
ref_pts = detectSURFFeatures(ref_img_gray); %Detecta puntos de interes.
[ref_features,ref_validPts]= extractFeatures(ref_img_gray,ref_pts);
%Extrae imagen en escala de grises los puntos de interés,
figure; imshow(ref_img); title('Imagen de referencia'); % Imprime la imagen RGB.
hold on; plot(ref_pts.selectStrongest(50));
%Imprime los puntos de interés sobre la imagen de referencia en RGB.

%Imagen de base detección de características.
base_pts = detectSURFFeatures(base_img_gray);
[base_features, base_validPts] = extractFeatures(base_img_gray, base_pts);
figure;imshow(base_img); title('Imagen de base');
hold on; plot(base_pts.selectStrongest(50));
%Imprime los puntos de interés sobre la imagen de base en RGB.
```

Figura 24. Script 7.

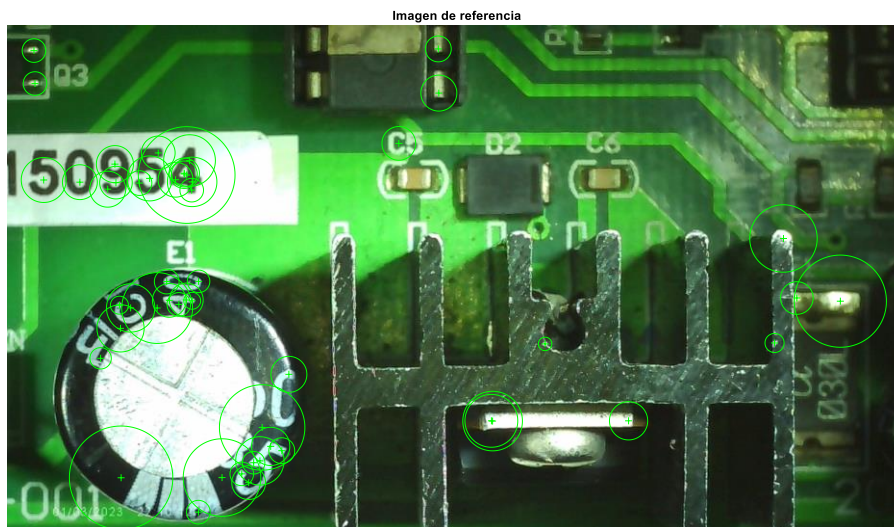


Figura 25. Puntos de interés en imagen de referencia.

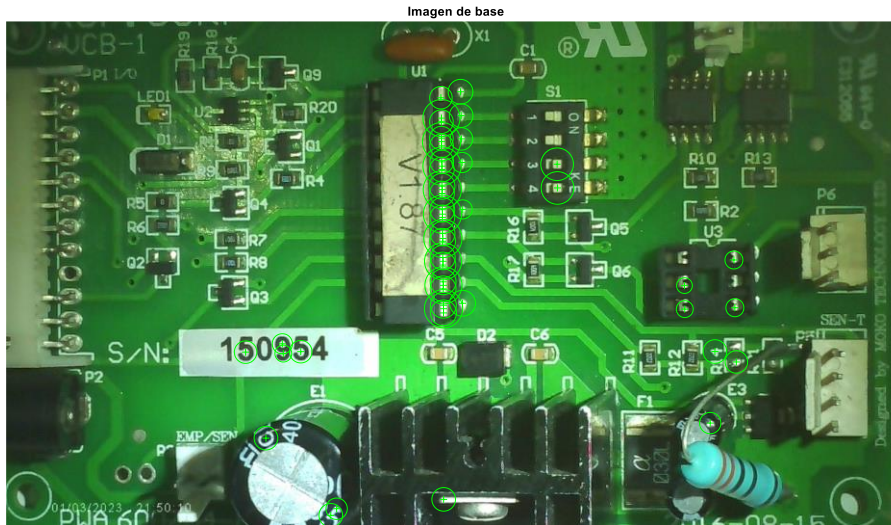


Figura 26. Puntos de interés en imagen de base.

Hallamos los puntos coincidentes entre la imagen de referencia y la imagen general.

Posteriormente se imprime se una sola imagen **Figura 28**.

Comparación de puntos de interés.

```

index_pairs = matchFeatures(ref_features, base_features);
%Encuentra puntos semejantes entre las dos imágenes.
ref_matched_pts = ref_validPts(index_pairs(:,1)).Location;
%Toma puntos de la imagen de referencia.
base_matched_pts = base_validPts(index_pairs(:,2)).Location;
%Toma puntos de la imagen base.
figure,showMatchedFeatures(base_img, ref_img,base_matched_pts,ref_matched_pts,'montage');
%Muestra las dos imágenes con los puntos semejantes.
title('Imprime todos los puntos');

```

Figura 27. Script 8.



Figura 28. Puntos semejantes entre la imagen de referencia y la imagen general.

Visualizamos los puntos realmente coincidentes en una sola imagen **Figura 30**, los cuales son un subconjunto de los puntos semejantes.

Comparación de puntos de interés.

```
[tform_matrix,inlierIdx]=estimateGeometricTransform2D(ref_matched_pts,base_matched_pts,'projective');  
%Encuentra solo los puntos realmente coincidentes.  
ref_inlier_pts = ref_matched_pts(inlierIdx,:); %Toma puntos de la imagen de referencia.  
base_inlier_pts = base_matched_pts(inlierIdx,:); %Toma puntos de la imagen de base.  
%Muestra las dos imágenes con los puntos realmente coincidentes.  
figure;showMatchedFeatures(base_img, ref_img, base_inlier_pts, ref_inlier_pts,'montage');  
title('Imprime solo los puntos realmente coincidentes');
```

Figura 29. Script 9.



Figura 30. Puntos realmente coincidentes.

Recomendaciones

Obtener la imagen de referencia y base de forma independiente y no hacer uso de herramientas de corte de imágenes.

PRACTICA #3

Tema

Detección y medición de micro objetos.

Objetivos

Objetivo general

- Implementar prácticas mediante software de procesamiento imágenes para la detección, medida de micro objetos y comparación de puntos de interés con un microscopio digital.

Objetivo específico

- Detectar y medir micro objetos de una imagen tomada con el microscopio digital.

Materiales y equipo

- Ordenador o laptop.
- Matlab.
- Microscopio digital ANNLOV.

Desarrollo

Detección y medición de micro objetos. Aplicamos el trabajo ya realizado en binarización de imágenes. Dividir la imagen a procesar en canales RGB y convertirlas a escala de grises, para así binarizar el componente que queramos utilizar.

Detección y medición de micro objetos.

```
clc;clear;
RGB= imread('santa3.jpeg'); %carga la imagen
```

Detección y medición de micro objetos.

```
f1 = figure(1); set(f1, 'color', 'w'); clf; % Crea Figura
subplot(1,3,1); % Selecciona el 1 subplot
imshow(RGB(:,:,1)); title('Componente Rojo'); % Visualiza componente R
subplot(1,3,2); % Selecciona el 2 subplot
imshow(RGB(:,:,2)); title('Componente Verde'); % Visualiza componente G
subplot(1,3,3); % Selecciona el 3 subplot
imshow(RGB(:,:,3)); title('Componente Azul'); % Visualiza componente B
```

Figura 31. Script 10.



Figura 32. Descomposición de componentes RGB.

Detección y medición de micro objetos.

```
R=RGB(:,:,1); %Matriz Roja
G=RGB(:,:,2); %Matriz Verde
B=RGB(:,:,3); %Matriz Azul
%Cálculo del umbral de binarización del componente en rojo en escala de grises.
nivel=graythresh(R);
red_santa=imbinarize(R,nivel); %Binarización de la matriz roja
figure (2)
imshowpair(R,red_santa,'montage');
%Imprime imagen del antes y después del componente en su etapa de binarización.
```

Figura 33. Script 11.

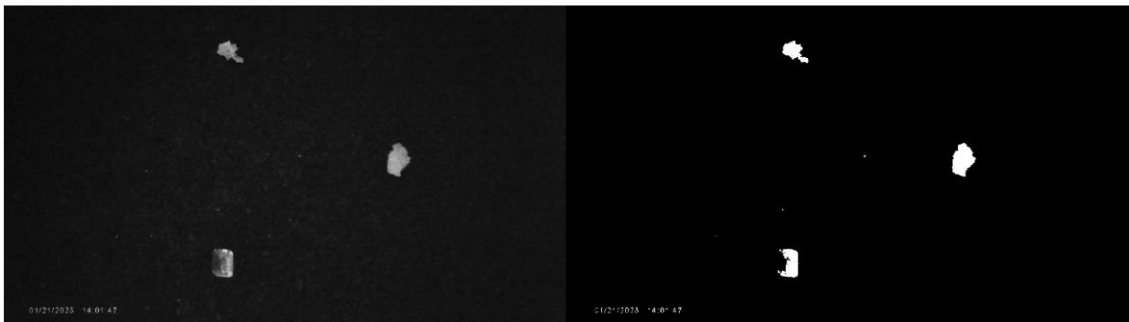


Figura 34. Imagen binarizada del componente en rojo.

Se crea un elemento estructurante en forma de disco con radio de 10 píxeles. Posteriormente se realiza una operación de cierre morfológico en la imagen [8]. Esto se aplica con la finalidad de preservar la magnitud del micro objeto en la misma.

Detección y medición de micro objetos.

```
bwm=bwareaopen(red_santa,200); %Aplicación del filtro  
io = strel('disk',10); %Elemento estructurante  
bwm = imclose(bwm,io); %Operación closing  
bwm=bwareaopen(bwm,350); %Aplicación del filtro
```

Figura 35. Script 12.

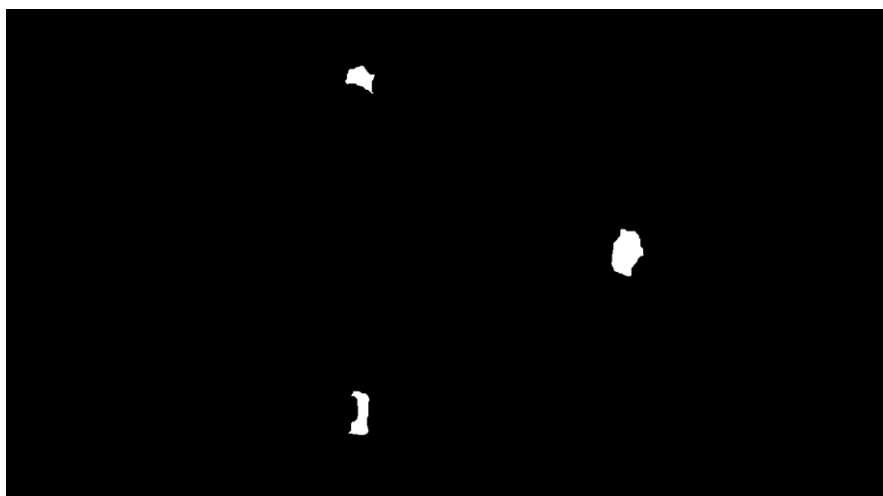


Figura 36. Binarización + filtro.

Encontramos micro objetos en la imagen binarizada, por medio de la creación de matrices. Posterior a esto guardamos las características generales de cada micro objeto encontrado. Las tablas de la **Figura 38**. las encontramos como variable en el workspace.

Detección y medición de micro objetos.

```
CC = bwconncomp(bwm); %Comando para encontrar elementos en una imagen binarizada
no=CC.NumObjects;
L = labelmatrix(CC); %Crea etiquetas para realizar la caracterización
[m,n]=size(L); %Halla las filas y columnas de la matriz L
figure(3)
imshow(bwm)
for i=1:no
    T(i)=i;
end
for i=1:m
    for j = 1:n
        if(L(i,j)~=T(1)) %Permite hallar solo la figura 1
            salida1(i,j) = 0;
        else
            salida1(i,j) =1;
        end
        if(L(i,j)~=T(2)) %Permite hallar solo la figura 2
            salida2(i,j) = 0;
        else
            salida2(i,j) =1;
        end
        if(L(i,j)~=T(3)) %Permite hallar solo la figura 3
            salida3(i,j) = 0;
        else
            salida3(i,j) =1;
        end
    end
end
end
```

Detección y medición de micro objetos.

```
primera=logical(salida1); %Etiqueta la primera imagen para hallar sus características
segunda=logical(salida2); %Etiqueta la segunda imagen para hallar sus características
tercera=logical(salida3); %Etiqueta la tercera imagen para hallar sus características

ESTADISTICAS1= regionprops(primera, 'all'); %Propiedades de la imagen 1
ESTADISTICAS2= regionprops(segunda, 'all'); %Propiedades de la imagen 2
ESTADISTICAS3= regionprops(tercera, 'all'); %Propiedades de la imagen 3
```

Figura 37. Script 13.

ESTADISTICAS1		ESTADISTICAS2		ESTADISTICAS3	
Field	Value	Field	Value	Field	Value
Area	2040	Area	1597	Area	2692
Centroid	[510.0711,585.5706]	Centroid	[518.6612,106.0056]	Centroid	[899.7348,351.3629]
BoundingBox	[486.5000,552.5000,...	BoundingBox	[490.5000,81.5000,...	BoundingBox	[875.5000,312.5000,...
SubarrayIdx	1x2 cell	SubarrayIdx	1x2 cell	SubarrayIdx	1x2 cell
MajorAxisLength	74.3264	MajorAxisLength	62.4342	MajorAxisLength	73.5165
MinorAxisLength	39.9368	MinorAxisLength	36.9636	MinorAxisLength	47.9413
Eccentricity	0.8434	Eccentricity	0.8059	Eccentricity	0.7581
Orientation	89.7649	Orientation	-37.4352	Orientation	76.5376
ConvexHull	46x2 double	ConvexHull	33x2 double	ConvexHull	51x2 double
ConvexImage	65x42 logical	ConvexImage	50x60 logical	ConvexImage	77x53 logical
ConvexArea	2465	ConvexArea	1938	ConvexArea	2905
Circularity	0.6658	Circularity	0.6008	Circularity	0.8162
Image	65x42 logical	Image	50x60 logical	Image	77x53 logical
FilledImage	65x42 logical	FilledImage	50x60 logical	FilledImage	77x53 logical
FilledArea	2040	FilledArea	1597	FilledArea	2692
EulerNumber	1	EulerNumber	1	EulerNumber	1
Extrema	8x2 double	Extrema	8x2 double	Extrema	8x2 double
EquivDiameter	50.9648	EquivDiameter	45.0928	EquivDiameter	58.5454
Solidity	0.8276	Solidity	0.8240	Solidity	0.9267
Extent	0.7473	Extent	0.5323	Extent	0.6596
PixelIdxList	2040x1 double	PixelIdxList	1597x1 double	PixelIdxList	2692x1 double
PixelList	2040x2 double	PixelList	1597x2 double	PixelList	2692x2 double
Perimeter	196.2260	Perimeter	182.7720	Perimeter	203.5890
PerimeterOld	204.4680	PerimeterOld	190.2670	PerimeterOld	211.4802
MaxFeretDiameter	69.3181	MaxFeretDiameter	68.2495	MaxFeretDiameter	77.0584
MaxFeretAngle	-116.5651	MaxFeretAngle	-140.9469	MaxFeretAngle	-92.2312
MaxFeretCoordinates	[524.5000,615.5000,...	MaxFeretCoordinates	[549.5000,130.5000,...	MaxFeretCoordinates	[904.5000,389.5000,...
MinFeretDiameter	41.0477	MinFeretDiameter	41.3772	MinFeretDiameter	51.6424
MinFeretAngle	2.3859	MinFeretAngle	-56.9761	MinFeretAngle	5.7106
MinFeretCoordinates	[528.5000,585.5000,...	MinFeretCoordinates	[533.5000,88.5000,...	MinFeretCoordinates	[928.5000,350.5000,...

Figura 38. Propiedades de los micro objetos.

Se filtra la imagen para obtener un resultado más preciso de los posibles micro objetos

Figura 40.

Detección y medición de micro objetos.

```
f4 = figure(4); set(f4, 'color', 'w'); clf; % Crea Figura
subplot(1,3,1); % Selecciona el 1 subplot
imshow(salida1); title('Objeto 1'); %
subplot(1,3,2); % Selecciona el 1 subplot
imshow(salida2); title('Objeto 2'); %
subplot(1,3,3); % Selecciona el 1 subplot
imshow(salida3); title('Objeto 3');
```

Figura 39. Script 14.

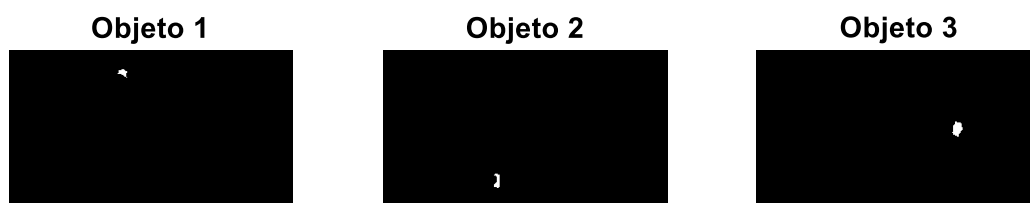


Figura 40. Objetos de forma individual.

Creamos etiquetas para cada objeto y utilizamos las características extraídas para posicionarlas de manera individual. Posterior a esto, se imprime el resultado **Figura 42**.

Detección y medición de micro objetos.

```
EJEX1=ESTADISTICAS1(1).Centroid(2); %Guarda el dato de centroide x del micro objeto 1
EJEY1=ESTADISTICAS1(1).Centroid(1); %Guarda el dato de centroide y del micro objeto 1
EJEX2=ESTADISTICAS2(1).Centroid(2); %Guarda el dato de centroide x del micro objeto 2
EJEY2=ESTADISTICAS2(1).Centroid(1); %Guarda el dato de centroide y del micro objeto 2
EJEX3=ESTADISTICAS3(1).Centroid(2); %Guarda el dato de centroide x del micro objeto 3
EJEY3=ESTADISTICAS3(1).Centroid(1); %Guarda el dato de centroide y del micro objeto 3

Lenght1=ESTADISTICAS1(1).MajorAxisLength;
%Guarda el dato de la longitud mayor del micro objeto 1
Lenght2=ESTADISTICAS2(1).MajorAxisLength;
%Guarda el dato de la longitud mayor del micro objeto 2
Lenght3= ESTADISTICAS3(1).MajorAxisLength;
%Guarda el dato de la longitud mayor del micro objeto 3

xt=[EJEX1,EJEX2,EJEX3]; %Coloca todos los centroides x en un vector
yt=[EJEY1,EJEY2,EJEY3]; %Coloca todos los centroides y en un vector
str={'Objeto 1','Objeto 2','Objeto 3'}; %Coloca el texto de cada par de centroides
figure(5),imshow(RGB)
text(yt,xt,str,'Color','w') %Se aplica el comando text a la imagen RGB inicial
```

Figura 41. Script 15.



Figura 42. Etiquetado de micro objetos.

Recomendaciones

Si los micro objetos presentes a procesar tienden a poseer un tono de color similar al de la base del microscopio. Ubique un lienzo (De preferencia negro) sobre ella, para no tener inconvenientes con el procesamiento de la imagen.

6. Conclusión

Matlab nos permite realizar una gran cantidad de tareas sobre el procesamiento de imágenes y análisis de datos. Lo que lo convierte en una software muy útil en este ámbito, el cual estamos incursionando como profesionales.

Con imágenes capturadas por nuestro microscopio digital, bajo ciertas condiciones expuestas en las prácticas y el uso de comandos adecuados para procesamiento de imágenes, podemos realizar diversas tareas como la detección - medición de micro objetos. Estas son las bases en procesos industriales, para el descarte de material no deseado.

Así mismo la detección de puntos de interés en imágenes de micro objetos, pueden ser usados para la detección de estos, reconocimiento de patrones, seguimiento/conductas, entre otras aplicaciones en diversos campos de estudio.

El trabajo en conjunto entre el microscopio digital ANNLOV y el software Matlab nos brindaran la oportunidad de captar e interpretar resultados precisos en el procesamiento de imágenes.

La implementación de las practicas, al igual que la incorporación del microscopio digital ANNLOV y sobre todo la colaboración de nuestros docentes, serán parte del estudio progresivo del procesamiento de imágenes en nuestra institución.

7. Cronograma y actividades para desarrollar

<u>ACTIVIDADES</u>	<u>NOV</u>			<u>DIC</u>			<u>ENE</u>			<u>FEB</u>		
	1-30			1-21			3-20			6-16		
<u>Realización del Documento</u>	■	■	■									
<u>Prueba y Error de los códigos</u>			■	■								
<u>Presentación de los Documentos al Tutor</u>				■	■							
<u>Corrección del Documento</u>					■							
<u>Compra del Microscopio Digital</u>						■						
<u>Prueba y Error de los códigos con imágenes</u>							■	■				
<u>Presentación al tutor</u>								■	■			
<u>Corrección Final del Documento</u>								■	■	■		
<u>Presentación Final del Documento</u>											■	■

Tabla 1. Cronograma.

8. Presupuesto

<u>Elementos</u>	<u>Costo</u>
<u>MICROSCOPIO DIGITAL</u>	<u>\$200</u>

Tabla 2. Presupuesto.

9. Bibliografía

- [Fernando Cortes, 27 Mayo 2014. [En línea]. Available:
1] <http://cortesfernando.blogspot.com/2014/05/binarizacion-imagen.html>.
- [Etitudela, «Etitudela,» [En línea]. Available:
2] <http://www.etitudela.com/celula/downloads/visionartificial.pdf>.

- [3] Chris Solomon, Toby Breckon, Fundamentals of Digital Image Processing
A Practical Approach with Examples in Matlab, The Atrium, Southern Gate,
Chichester,; 9600 Garsington Road, Oxford, OX4 2DQ, UK, 2011.
- [4] Leica Camera, «Leica Camera,» [En línea]. Available: <https://www.leica-microsystems.com/es/productos/microscopios-digitales/>.
- [5] ANNLOV, «ANNLOV,» [En línea]. Available: <https://www.annlov.net/7-inch-lcd-digital-microscope-with-32gb-tf-card-50-1200x-magnification-1080p-usb-video-microscope-camera-with-wired-remote-for-circuit-board-soldering-pcb-kids-coin-microscope-p00100p1.html>. [Último acceso: 30 12 2022].
- [6] O. Nobuyuki, «IEEE,» 1 1 1979. [En línea]. Available:
<https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4310076&isnumber=4310064>. [Último acceso: 10 2 2023].
- [7] L. D. P. G. Daniel Santos, «scielo,» 20 12 2020. [En línea]. Available:
<https://www.scielo.cl/pdf/infotec/v31n3/0718-0764-infotec-31-03-23.pdf>. [Último acceso: 12 2 2023].
- [8] MathWorks, «MathWorks,» [En línea]. Available:
<https://la.mathworks.com/help/images/morphological-dilation-and-erosion.html>.
[Último acceso: 16 2 2023].