

## **1. Introducción**

A través de los años el hombre ha presentado un cambio radical en su nivel de vida; los conocimientos que él ha logrado acumular y aplicar en su trabajo han sido para su beneficio, que ha cambiado radicalmente su modo de vivir.

El relativo reciente desarrollo de la tecnología nos ha hecho posible vivir en algunos aspectos, mejor de lo que podíamos pensar en el pasado. Algunas de estas mejoras han sido tan sutiles, que han pasado desapercibidas y quizá no le demos la importancia que debería.

Uno de los elementos más importantes de nuestra vida moderna es sin duda la computadora. Esta ha venido a simplificar nuestra existencia de muchas maneras. Las agencias gubernamentales, la empresa privada, las instituciones educativas y otras entidades utilizan las computadoras para llevar a cabo transacciones, automatizar procesos, enseñar o sencillamente con fines de entretenimiento. Esta es también una herramienta que ha venido a acortar distancias por medio de la comunicación. El uso de la computadora ha mejorado y agilizado muchas de nuestras labores diarias que realizamos tanto en el hogar como en el trabajo.

Tras el desarrollo de las primeras computadoras surgió la necesidad de programarlas para que realizaran las tareas deseadas. Un lenguaje de programación es un lenguaje formal diseñado para expresar procesos que pueden ser llevados a cabo por máquinas como las computadoras. Pueden usarse para crear programas que controlen el comportamiento físico y lógico de una máquina, para expresar algoritmos con precisión, o como modo de comunicación humana. Está formado por un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones. Al proceso por el cual se escribe, se prueba, se depura, se compila (de ser necesario) y se mantiene el código fuente de un programa informático se le llama programación.

Los lenguajes de programación han evolucionado a la par de la tecnología que envuelve a las computadoras, a tal grado que se han desarrollado a lo que se ha dado llamar Inteligencia Artificial (IA), que es una combinación de la ciencia del computador, fisiología y filosofía, tan general y amplio como eso, es que reúne varios campos (robótica, sistemas expertos, por ejemplo), todos los cuales tienen en común la creación de máquinas que pueden pensar.

La Inteligencia Artificial trata de conseguir que los ordenadores simulen en cierta manera la inteligencia humana. Se acude a sus técnicas cuando es necesario incorporar en un sistema informático, conocimiento o características propias del ser humano.

La visión por computadora es una rama de la inteligencia artificial que tiene por objetivo modelar matemáticamente los procesos de percepción visual en los seres vivos y generar programas que permitan simular estas capacidades visuales por computadora

Dentro de la visión computacional, podemos encontrar el procesamiento de imágenes, y dentro de este, una parte muy importante se encarga del análisis de las mismas. Esto es, dada una imagen, lo que deseamos obtener es una descripción de dicha imagen. El análisis depende primeramente de detectar determinadas partes de la imagen (regiones u objetos). Para generar tal descripción es necesario segmentar adecuadamente e identificar la región deseada.

La segmentación en el campo de la visión artificial es el proceso de dividir una imagen digital en varias partes (grupos de píxeles) u objetos. El objetivo de la segmentación es simplificar y/o cambiar la representación de una imagen en otra más significativa y más fácil de analizar. La segmentación se usa tanto para localizar objetos como para encontrar los límites de estos dentro de una imagen. Más precisamente, la segmentación de la imagen es el proceso de asignación de una etiqueta a cada píxel de la imagen de forma que los píxeles que compartan la misma etiqueta también tendrán ciertas características visuales similares.

En el presente trabajo se plantea el desarrollo de un algoritmo y programa para la obtención de un área específica de una imagen digital obtenida de forma convencional en hojas de frutos con la herramienta Guide de MatLab. El objetivo es extraer las características de dicha imagen digital haciendo uso del algoritmo con la finalidad de cuantificar el crecimiento enzimático.

## **2. Justificación**

En uno de los laboratorios del Instituto Tecnológico de Tuxtla Gutiérrez se desarrollan proyectos de investigación. Uno de los cuales consiste en analizar el crecimiento enzimático en hojas frutales, para lo cual el investigador de forma directa y basado en su experiencia evalúa el crecimiento mediante un muestreo sucesivo a diferentes horas del día. Ocasionalmente el investigador debe dar continuidad a este muestreo durante la noche lo cual lo hace un proceso cansado.

Con el sistema de visión que se pretende desarrollar le facilitara al investigador analizar el crecimiento enzimático mediante el procesamiento de imágenes digitales obtenidas a través de una cámara convencional.

## **3. Objetivos**

### **3.1. Objetivo General**

Desarrollar un sistema de visión para el reconocimiento del crecimiento de hongos y cuantificar el crecimiento enzimático mediante el procesamiento de imágenes digitales de hoja de frutos obtenidas a través de una cámara convencional usando el Toolbox de procesamiento de imágenes de MatLab.

### **3.2. Objetivos específicos.**

- Desarrollo de algoritmo para procesamiento de imágenes.
- Desarrollo de software para el procesamiento de imágenes.
- Cuantificar el crecimiento enzimático mediante el conteo de pixeles.
- Segmentación de la imagen.
- Pruebas de filtrado y binarización a la imagen digital.

#### 4. Cronograma de actividades realizadas

Actividad	Semana															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Introducción de Sistemas de visión																
Desarrollo de Algoritmo de procesamiento de imágenes																
Desarrollar interfaz hombre-máquina para el procesamiento de imágenes																
Captar y registrar datos en el sistema desarrollado																
Informe de avance de proyecto																
Presentación de resultados y reporte final																

#### 4.1. Descripción detallada de las actividades

- a) Búsqueda de la información a fin al tema para tomar referencias y novedades.
- b) Desarrollo de interfaz gráfica de usuario en MATLAB para procesamiento de imágenes
- c) Pruebas del algoritmo de procesamiento de imágenes, utilizando la interfaz de usuario y una computadora personal.
- d) Informe de avance del proyecto y reporte final como lo marca el programa se entregará un informe parcial de las actividades desarrolladas, en la residencia así como el avance, problemática encontrada y estrategias para terminar a tiempo. Entrega del Informe final que deberá incluir:

- Portada
- Índice
- Introducción
- Justificación
- Objetivos
- Caracterización del área en que participo
- Problemas a resolver, priorizándolos.
- Alcances y limitaciones
- Fundamento teórico
- Procedimiento y descripción de las actividades realizadas.
- Resultados, planos, graficas, prototipos y programas.
- Conclusiones y recomendaciones.
- Referencias bibliográficas.

## **5. Caracterización del área en que participó**

### **5.1. Historia del Instituto Tecnológico de Tuxtla Gutiérrez**

El Instituto Tecnológico de Tuxtla Gutiérrez es una institución pública dependiente de la Secretaría de Educación Pública. Imparte 8 licenciaturas y 2 programas de posgrado en las áreas de Ingeniería, Tecnología y Ciencias Administrativas.

El Instituto Tecnológico de Tuxtla Gutiérrez fue fundado el 24 de octubre de 1972, se encuentra ubicado en el centro del estado de Chiapas, que por su gran variedad de climas y suelos es propio para el cultivo de muy diversas especies vegetales nativas y adaptadas, con lo que se puede desarrollar la agroindustria, así como muchos otros procesos industriales a partir del gran potencial que ofrece este estado, en el cual el Instituto debe constituirse en el sujeto de cambio, al presentar alternativas de desarrollo sustentable del estado en forma multidisciplinaria.

La matrícula escolar en licenciatura se conforma de 1 982 estudiantes: 1 473 hombres y 509 mujeres. En el posgrado hay 78 alumnos: 68 hombres y 10 mujeres.

La población de egresados en el nivel de licenciatura es de 2 345 alumnos, de ellos 1 892 son hombres y 453 son mujeres.

Los principales laboratorios con los que cuenta el Instituto Tecnológico de Tuxtla Gutiérrez son:

- Microbiología
- Biotecnología
- Química
- Química pesada
- Mecánica

- Sistemas computacionales
- Ingeniería industrial
- Plantas piloto
- Polo Tecnológico Nacional

Proyectos de vinculación exitosos con la industria

- Aprovechamiento integral del timbre
- Incremento en el contenido de sacarosa en la caña de azúcar
- Efectos del boro en la producción del cacahuete
- Factibilidad técnica en la extracción de aceites esenciales de la flora chiapaneca

#### **5.1.1. Misión**

Formar de manera integral profesionistas de excelencia en el campo de la ciencia y la tecnología con actitud emprendedora, respeto al medio ambiente y apego a los valores éticos.

#### **5.1.2. Visión**

Ser una Institución de excelencia en la educación superior tecnológica del Sureste, comprometida con el desarrollo socioeconómico sustentable de la región.

### **5.2. Área específica relacionada directamente con el proyecto**

En el área de Ing. Bioquímica se cuenta con tres invernaderos y cuatro laboratorios: Biotecnología, Investigación, Analítica, Cultivo de Tejidos Vegetales y además 11 laboratorios del Polo Tecnológico Nacional para el desarrollo de investigación y pruebas analíticas en biocombustibles con equipos nuevos, adquiridos principalmente entre los años 2005 y 2012 mediante la participación en diferentes convocatorias emitidas por diversas

entidades públicas y privadas, acorde con las líneas de investigación del programa. Todos los laboratorios cuentan con equipos de seguridad contra incendios.

## **6. Alcances y limitaciones**

### **6.1. Alcances**

- Herramienta de software diseñada para procesar las imágenes digitales haciendo uso del algoritmo.

### **6.2. Limitaciones**

- El proyecto se limita a las imágenes ya proporcionadas.
- Tiempo de espera de un lapso de 6 semanas para que sean proporcionadas nuevas imágenes debido al deterioro de la muestra.
- Cámara no especializada para la adquisición de imágenes.

## **7. Fundamento teórico**

### **7.1. MatLab**

MATLAB (abreviatura de MATrix LABoratory, "laboratorio de matrices") es una herramienta de software matemático que ofrece un entorno de desarrollo integrado (IDE) con un lenguaje de programación propio (lenguaje M). Está disponible para las plataformas Unix, Windows, Mac OS X y GNU/Linux.

Entre sus prestaciones básicas se hallan: la manipulación de matrices, la representación de datos y funciones, la implementación de algoritmos, la creación de interfaces de usuario (GUI) y la comunicación con programas en otros lenguajes y con otros dispositivos hardware. El paquete MATLAB dispone de dos herramientas adicionales que expanden sus prestaciones, a saber, Simulink (plataforma de simulación multidominio) y GUIDE (editor

de interfaces de usuario - GUI). Además, se pueden ampliar las capacidades de MATLAB con las cajas de herramientas (toolboxes); y las de Simulink con los paquetes de bloques (blocksets).

### **7.1.1. Origen**

MATLAB nace como una solución a la necesidad de mejores y más poderosas herramientas de cálculo para resolver problemas de cálculo complejos en los que es necesario aprovechar las amplias capacidades de proceso de datos de grandes computadores.

El nombre MATLAB viene de "matrix laboratory" (laboratorio matricial). MATLAB fue originalmente escrito para proveer acceso fácil al software matricial desarrollado por los proyectos LINPACK y EISPACK, que juntos representan el estado del arte e software para computación matricial. Hoy MATLAB es usado en una variedad de áreas de aplicación incluyendo procesamiento de señales e imágenes, diseño de sistemas de control, ingeniería financiera e investigación médica. La arquitectura abierta facilita usar MATLAB y los productos que lo acompañan para explorar datos y crear herramientas personalizadas que proveen visiones profundas tempranas y ventajas competitivas.

### **7.1.2 Image Processing Toolbox**

Image Processing Toolbox™ proporciona un conjunto completo de algoritmos estándar de referencia, funciones y aplicaciones para el procesamiento, el análisis y la visualización de imágenes, así como para el desarrollo de algoritmos. Puede realizar mejoras de imágenes, enfoque de imágenes borrosas, detección de funciones, reducción de ruidos, segmentación de imágenes, transformaciones geométricas y registro de imágenes. Muchas de las funciones de la toolbox son multiproceso, para aprovechar los ordenadores con varios núcleos o procesadores.

Image Processing Toolbox soporta un conjunto diverso de tipos de imágenes, tales como las de alto rango dinámico, las de resolución de gigapíxeles, las de perfiles ICC embebidos y las tomográficas. Las funciones de visualización permiten explorar una imagen, examinar una región de píxeles, ajustar el contraste, crear contornos o histogramas y manipular regiones de interés (ROIs). Con los algoritmos de la toolbox puede restaurar imágenes degradadas, detectar y medir funciones, analizar formas y texturas y ajustar el balance de color.

#### **7.1.2.1. Funciones principales**

- Mejora y filtrado de imágenes y enfoque de imágenes borrosas.
- Análisis de imágenes, incluyendo segmentación, morfología, extracción de funciones y medición.
- Transformaciones geométricas y métodos de registro de imágenes basados en intensidad.
- Transformaciones de imágenes, incluyendo FFT, DCT, Radon y proyección de haz de rayos en abanico.
- Flujos de trabajo para procesar, visualizar y navegar por imágenes arbitrariamente grandes.
- Aplicaciones Visor de imágenes y Visor de vídeos.
- Importación y exportación de archivos DICOM.

#### **7.1.2.2. Importar y exportar imágenes**

Image Processing Toolbox soporta imágenes generadas por un amplio gama de dispositivos, incluidas cámaras digitales, sensores satelitales y aéreos, dispositivos de generación de imágenes médicas, microscopios, telescopios y otros instrumentos científicos. Puede visualizar, analizar y procesar estas imágenes en muchos tipos de datos, incluidos los enteros de punto flotante de precisión simple y doble, y los enteros de 8, 16 y 32 bits con signo y sin signo.

Existen varias formas de importar y exportar las imágenes en el entorno de MATLAB® para su procesamiento. Puede utilizar Image Acquisition Toolbox™ para adquirir imágenes en vivo de cámaras web, capturadoras de vídeo, cámaras DCAM, cámaras GigE Vision y otros dispositivos. Mediante Database Toolbox™ puede acceder a imágenes almacenadas en bases de datos conformes con ODBC o JDBC.

### **7.1.2.3. Formatos estándar y especializados soportados**

MATLAB soporta formatos de datos e imágenes estándar, como JPEG, JPEG-2000, TIFF, PNG, HDF, HDF-EOS, FITS, Microsoft® Excel®, ASCII, y archivos binarios. También soporta los formatos de imágenes multibanda BIP y BIL utilizados por LANDSAT. La E/S de bajo nivel y las funciones de asignación de memoria le permiten desarrollar rutinas personalizadas para trabajar con cualquier tipo de datos.

Image Processing Toolbox soporta diversos formatos de archivo de imagen especializados. En el caso de las imágenes médicas, soporta los archivos DICOM, incluidos los metadatos asociados, así como los formatos Analyze 7.5 e Interfile. La toolbox también puede leer imágenes geoespaciales en archivos NITF e imágenes de alto rango dinámico en archivos HDR.

### **7.1.2.4. Visualizar imágenes**

Image Processing Toolbox proporciona funciones de visualización de imágenes que son altamente personalizables. Puede crear pantallas con varias imágenes en una sola ventana, realizar anotaciones en las pantallas con texto y gráficos y crear pantallas especializadas, tales como histogramas, perfiles y diagramas de contorno.

Esta toolbox incluye una aplicación para visualizar vídeos y secuencias. Las funciones de visualización de volumen de MATLAB permiten crear visualizaciones de isosuperficies de conjuntos de datos de imágenes multidimensionales.

#### **7.1.2.5. Explorar imágenes**

Además de funciones de visualización, la toolbox proporciona un conjunto de funciones para explorar imágenes. Puede ver información de la imagen, aplicarle el zoom y obtener panorámicas de la misma, al igual que examinar detenidamente una región de píxeles. Puede ubicar y manipular ROI de manera interactiva, incluyendo puntos, líneas, rectángulos, polígonos, elipses y formas a mano libre. También puede recortar, ajustar el contraste y medir distancias de manera interactiva. El conjunto de herramientas se encuentra disponible dentro de la Image Tool o mediante funciones individuales que se pueden utilizar para crear interfaces personalizadas.

#### **7.1.2.6. Mejora de la imagen**

Las técnicas de mejora de imágenes de Image Processing Toolbox le permiten aumentar la proporción de señal a ruido y acentuar las características de las imágenes modificando los colores o las intensidades de una imagen. Puede:

- Realizar ecualización de histogramas.
- Efectuar estiramiento de decorrelación.
- Reasignar el rango dinámico.
- Ajustar el valor gamma.
- Realizar filtrado lineal, medio o adaptativo.

La toolbox incluye rutinas de filtrado especializadas y una función de filtrado multidimensional generalizado que maneja tipos de imágenes de enteros, ofrece opciones de

relleno de contornos múltiples y realiza convolución y correlación. También se proporcionan filtros y funciones para diseñar e implementar sus propios filtros lineales.

#### **7.1.2.7. Enfoque de imágenes borrosas**

Los algoritmos de enfoque de imágenes borrosas de Image Processing Toolbox incluyen la deconvolución ciega, de Lucy-Richardson, de Wiener y de filtros regularizados, además de conversiones entre la distribución de puntos y funciones de transferencia óptica. Estas funciones ayudan a corregir el enfoque de imágenes borrosas por óptica fuera de foco, movimiento de la cámara o de la persona durante la captura de la imagen, condiciones atmosféricas, tiempo de exposición breve y otros factores. Todas las funciones de enfoque de imágenes borrosas funcionan con imágenes multidimensionales.

#### **7.1.2.8. Administración del color independiente del dispositivo**

La administración del color independiente del dispositivo de Image Processing Toolbox le permite representar el color con precisión independientemente del dispositivo de entrada o salida. Esto le ayuda a analizar las características de un dispositivo, medir la precisión del color cuantitativamente o desarrollar algoritmos para varios dispositivos distintos. Con las funciones especializadas de la toolbox, puede convertir imágenes entre espacios de color independientes del dispositivo, tales como sRGB, XYZ, xyY,  $L^*a^*b^*$ , uvL y  $L^*ch$ .

#### **7.1.2.9. Transformación de la imagen**

Las transformaciones de imágenes tales como FFT y DCT desempeñan un papel fundamental en muchas tareas de procesamiento de imágenes, como la mejora, el análisis, la restauración y la compresión de imágenes. Image Processing Toolbox brinda diversas transformaciones de imágenes, incluidas las proyecciones Radon y de proyección de haz de rayos en abanico. Puede reconstruir imágenes a partir de datos de proyección de haz de rayos en abanico y en paralelo (común en aplicaciones tomográficas). Las transformaciones de imágenes también están disponibles en MATLAB y Wavelet Toolbox™.

#### **7.1.2.10. Conversión de la imagen**

Las conversiones de imágenes entre clases de datos y tipos de imágenes son un requisito común para las aplicaciones de generación de imágenes. Image Processing Toolbox proporciona diversas utilidades para la conversión entre clases de datos, incluidos los enteros de punto flotante de precisión simple y doble, y los enteros de 8, 16 y 32 bits con signo y sin signo. La caja de herramientas incluye algoritmos para la conversión de tipos de imágenes, incluidas las imágenes binarias, en escala de grises, de color indexado y de color verdadero (truecolor). Para las imágenes en color en concreto, la toolbox admite diversos espacios de color, tales como YIQ, HSV y YCrCb, así como imágenes Bayer codificadas por patrones y de alto rango dinámico.

#### **7.1.2.11. Algoritmos de detección de bordes**

Los algoritmos de detección de bordes le permiten identificar contornos de objetos en una imagen. Estos algoritmos incluyen los métodos gaussianos de Sobel, Prewitt, Roberts, Canny y Laplacian. El potente método Canny puede detectar bordes muy débiles sin verse afectado por el ruido.

#### **7.1.2.12. Algoritmos de segmentación de imágenes**

Los algoritmos de segmentación de imágenes determinan los límites o bordes de la región en una imagen. Puede explorar varios métodos diferentes para la segmentación de imágenes, entre los que se incluyen la creación automática de umbrales, métodos basados en bordes y métodos morfológicos, como la transformada Watershed, que se utiliza habitualmente para segmentar objetos en contacto.

### **7.1.2.13. Operadores morfológicos**

Los operadores morfológicos le permiten detectar bordes, mejorar el contraste, eliminar el ruido, segmentar una imagen en regiones, reducir regiones o llevar a cabo su esqueletización. Las funciones morfológicas de Image Processing Toolbox incluyen:

- Erosión y dilatación.
- Apertura y cierre.
- Etiquetado de componentes conectados.
- Segmentación Watershed.
- Reconstrucción.
- Transformación de distancias.

### **7.1.2.14. Análisis de imágenes avanzado**

Image Processing Toolbox también contiene funciones de análisis de imágenes avanzadas que le permiten:

- Medir las propiedades de una determinada región de la imagen, como el área, el centro de masa o el cuadro de límites.
- Detectar líneas y extraer segmentos de líneas de una imagen mediante la transformada de Hough.
- Medir propiedades, como la rugosidad o la variación de color de una superficie, mediante funciones de análisis de texturas

### **7.1.3. GUIDE**

La interfaz gráfica de usuario, conocida también como GUI (del inglés graphical user interface) es un programa Informático que actúa de interfaz de usuario, utilizando un conjunto de imágenes y objetos gráficos para representar la información y acciones disponibles en la interfaz. Su principal uso, consiste en proporcionar un entorno visual sencillo para permitir la comunicación con el sistema operativo de una máquina o computador.

Habitualmente las acciones se realizan mediante manipulación directa, para facilitar la interacción del usuario con la computadora. Surge como evolución de las interfaces de línea de comandos que se usaban para operar los primeros sistemas operativos y es pieza fundamental en un entorno gráfico. Como ejemplos de interfaz gráfica de usuario, cabe citar los entornos de escritorio Windows, el X-Window de GNU/Linux o el de Mac OS X, Aqua.

En el contexto del proceso de interacción persona-ordenador, la interfaz gráfica de usuario es el artefacto tecnológico de un sistema interactivo que posibilita, a través del uso y la representación del lenguaje visual, una interacción amigable con un sistema informático.

GUIDE es un entorno de programación visual disponible en MATLAB para realizar y ejecutar programas que necesiten ingreso continuo de datos.

## **8. Resultados, planos, gráficas, prototipos y programas**

### **8.1. Algoritmo desarrollado**

```
%% Cargar imagen y recortar  
figure(1)  
Imag=imread('DSC00622.JPG');  
I=imcrop(Imag);
```

```

I=rgb2gray(I);
subplot(2,1,1),imshow(I);
title('Imagen recortada');

subplot(2,1,2),imhist(I);
title('Histograma');

%% Ajuste de Imagen (Aplicacion de filtros)
figure(2)

h1=fspecial('gaussian');
I=imfilter(I,h1);
subplot(2,2,1),imshow(I);
title('Filtro Gaussiano');

a=imadjust(I,stretchlim(I),[]);
subplot(2,2,2),imshow(a);
title('Imagen ajustada con imadjust');%Ajuste del contraste

a=adapthisteq(a);
subplot(2,2,3),imshow(a);
title('Imagen ajustada con adapthiteq');

a=medfilt2(a,[5 5]);
subplot(2,2,4),imshow(a);
title('Imagen con filtro Mediana de 5x5');

%% Binarizacion de Imagen (Aplicacion de filtros)
figure(3)

```

```

umbral = graythresh(a);
BW = im2bw(a,umbral);
BW = bwareaopen(BW, 500);
subplot(2,2,1),imshow(BW);
title('Imagen binarizada con pixeles borrados menores de 500');

```

```

% Aplicacion de filtros
BW2=bwmorph(BW,'open',4);% Eliminar picos
BW2=bwmorph(BW2,'close',4);% Eliminar huecos
subplot(2,2,2),imshow(BW2);
title('Imagen resultante con filtro 1)open, 2)close');

```

```

% Dilatación
B=strel('line',3,35);
B1=strel('line',3,45);
I2=imdilate(BW2,[B B1],3);
subplot(2,2,3),imshow(I2);
title('Imagen dilatada');

```

```

I2=imfill(I2,'holes');
subplot(2,2,4),imshow(I2);
title('Relleno de imagen para eliminar ruido');

```

```

%% Etiquetado de imagenes
figure(4)
[im Ne]=bwlabel(~I2);
imshow(label2rgb(im));

```

```

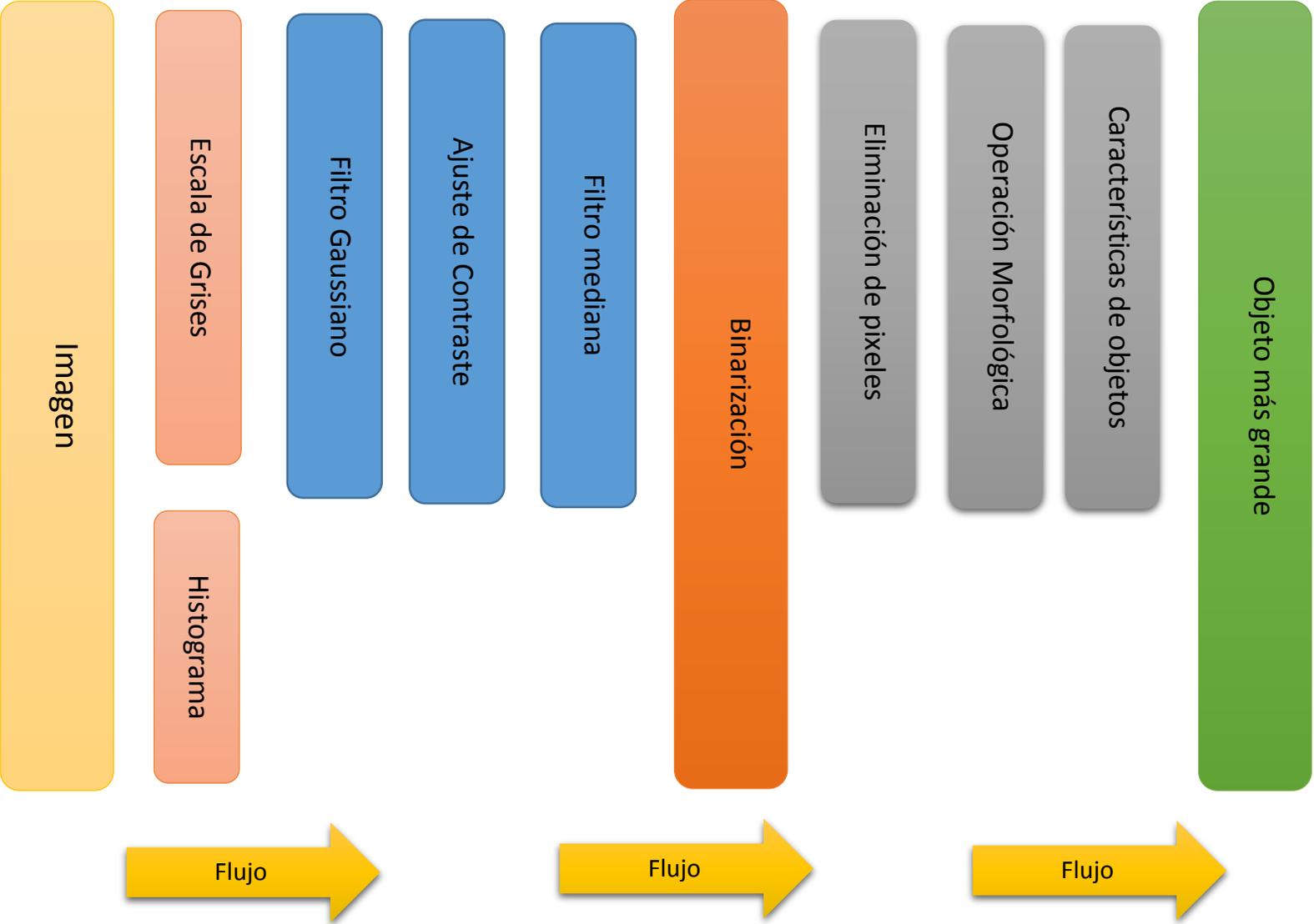
prop=regionprops(im);
hold on;

```

```
for n=1:length(prop)
    rectangle('Position',prop(n).BoundingBox,'EdgeColor','g','LineWidth',2);
    x=prop(n).Centroid(1);
    y=prop(n).Centroid(2);
    plot(x,y,'*');
end
hold off
```

```
%% Seleccionar objeto más grande
figure(5);
areas=[prop.Area];
[area_max pam]=max(areas);
am=ismember(im,pam);
prop=regionprops(am);
imshow(am);
title('Imagen obtenida');
xlabel(prop.Area);
```

8.2. Matemática del algoritmo



### 8.2.1. Obtención de imagen.

En este paso se carga la imagen al área de trabajo de MatLab como variable para manejarlo de manera matricial al aplicar diferentes filtros.

### 8.2.2 Escala de grises.

Los histogramas son distribuciones que describen la frecuencia con la que se presentan los valores de intensidad (píxeles) de la imagen. En el caso más sencillo los histogramas son mejor entendidos por medio de imágenes a escala de grises. Para una imagen a escala de grises  $I(u, v)$  con intensidades en el intervalo  $[0, K - 1]$  contendrá el histograma  $H$  exactamente  $K$  diferentes valores, el cual considerando una típica imagen a escala de grises de 8 bits, será  $H = 2^8 = 256$ . Cada valor del histograma es definido como  $h(i) = a$  el número de píxeles de  $I$  con el valor de intensidad  $i$  para todos los valores  $0 \leq i < K$ . Esto expresado formalmente:

$$h(i) = \text{card}^2\{(u, v) | I(u, v) = i\} \quad \text{Ec. 8.1}$$

$h(0)$  es entonces el número de píxeles con el valor 0,  $h(i)$  el número de píxeles que tienen el valor de 1, sucesivamente mientras que finalmente  $h(255)$  representa el número de píxeles blancos (con el máximo valor de intensidad) de la imagen. Como el resultado del cálculo del histograma se obtiene un vector unidimensional  $h$  con una longitud de  $K$ .

### 8.2.3 Filtrado Gaussiano.

La función de transferencia del filtro gaussiano de paso alto con frecuencia de corte a una distancia  $D_0$  del origen está dado por:

$$H_{hp} = 1 - e^{-D^2(u,v)/2D_0^2} \quad \text{Ec. 8.2}$$

Esta ecuación se deduce directamente de las ecuaciones. (8.3) y (8.4). Como era de esperar, los resultados obtenidos son más suaves que con los dos filtros anteriores. Incluso el filtrado de los objetos pequeños y barras delgadas es más limpio con el filtro gaussiano.

$$H_{hp}(u, v) = 1 - H_{Ip}(u, v) \quad \text{Ec. 8.3}$$

$$H_{hp} = e^{-D^2(u,v)/2D^2_0} \quad \text{Ec. 8.4}$$

#### 8.2.4 Contraste e iluminación o brillo.

Por contraste en una imagen podemos definirlo como la relación entre los diferentes valores de intensidad presentes en una imagen. La iluminación o brillo por su parte se relaciona con la forma en que los valores de intensidad se distribuyen, de tal forma que si estos se concentran más hacia valores pequeños de intensidad la imagen se apreciara más oscura, por el contrario si los valores de intensidad se concentran hacia los valores altos de intensidad la imagen se apreciara brillante o iluminada. Para ejemplificar estas operaciones pueden ser citados los ejemplos de aumentar el contraste en un 50% de una imagen, que equivaldría a tener una función homogénea que multiplique al pixel por 1.5, o bien elevar la iluminación o brillo en 10 niveles, que equivaldría a tener una función que e suma 10 al pixel en cuestión. Con lo que las funciones homogéneas quedarían definidas como:

$$f_c(I(x, y)) = I(x, y) \cdot 1.5 \text{ y } f_h(I(x, y)) = I(x, y) + 10 \quad \text{Ec. 8.5}$$

El operador genérico  $f(\cdot)$  que se utiliza para modificar el contraste o la iluminación en una imagen puede ser definido como:

$$I(x, y) = f(x, y) = c \cdot I(x, y) + b \quad \text{Ec 8.6}$$

Donde  $c$  modifica el valor del contraste mientras que  $b$  el del brillo o iluminación.

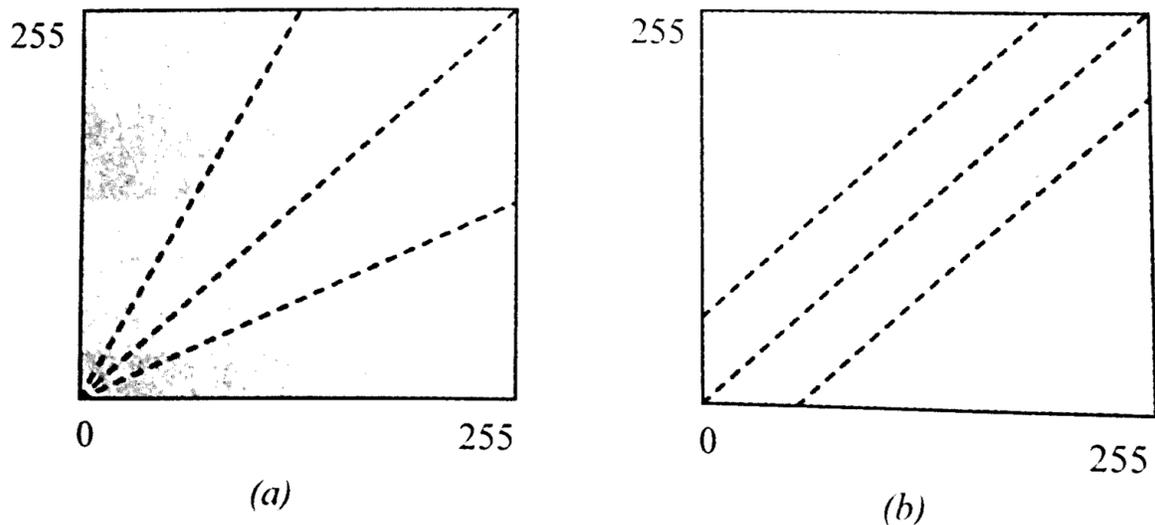


Fig. 8.1. Representación gráfica del mapeo que se genera en el pixel resultado  $I'(x, y)$ .

### 8.2.5 Filtro mediana.

En estadística la mediana es el valor de la variable que deja el mismo número de datos antes y después que él. De acuerdo con esta definición el conjunto de datos menores o iguales que la mediana representaran el 50% de los datos, y los que sean mayores que la mediana representaran el otro 50% del total de datos de la muestra.

Considerando que  $x_1, x_2, \dots, x_n$  son los datos de una muestra ordenada en orden creciente, la mediana quedaría definida como:

$$M_e = \frac{x_{n+1}}{2} \quad \text{Ec. 8.7}$$

Si  $n$  es impar será  $M_e$  la observación central de los valores, una vez que estos han sido ordenados en orden creciente o decreciente, Si  $n$  es par será el promedio aritmético de las dos observaciones centrales, esto es:

$$M_e = \frac{\left(\frac{x_n}{2} + \frac{x_{n+1}}{2}\right)}{2} \quad \text{Ec. 8.7}$$

Con lo anteriormente visto se puede decir que el filtro de la mediana sustituye cada pixel de la imagen por la mediana de los valores de intensidad dentro de la región de influencia  $R(x,y)$  definida por el filtro, esto expresado formalmente quedaría como:

$$I'(x,y) = M_e(R(x,y)) \quad \text{Ec. 8.8}$$

Para el cálculo de la mediana de los datos que conforman la región de interés  $R(x,y)$ , solo es necesario realizar dos pasos. Primero acomodar los valores de intensidad de la imagen que corresponden a la región de interés definida por el filtro en forma de vector, después reacomodarlos en orden creciente, si existen valores repetidos también serán repetidos en el nuevo arreglo.

Debido a que un filtro normalmente se define como una matriz cuyo conjunto de datos es impar, el valor de la mediana corresponde siempre al valor central del vector ordenado en forma creciente correspondiente a la región de interés  $R(x,y)$ .

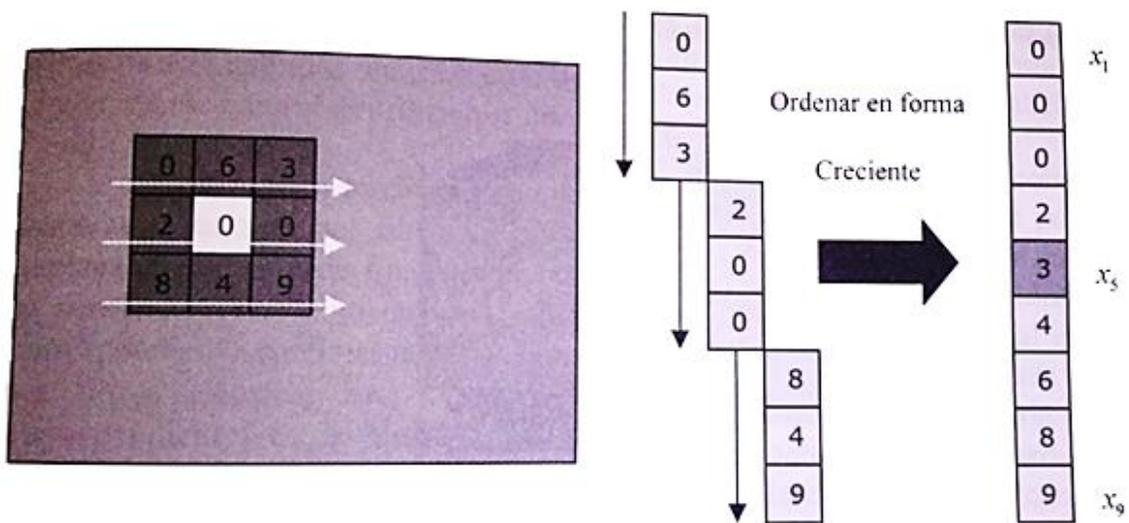


Fig. 8.2 Cálculo de la mediana considerando un filtro de tamaño 3X3.

El filtro de la mediana puede decirse que no produce o calcula el nuevo valor con el que será sustituido el pixel de la imagen, sino que este es un valor existente en la región de interés del filtro producto del reacomodo de datos.

### 8.2.6 Binarización.

La umbralización es una técnica de segmentación ampliamente utilizada en las aplicaciones industriales. Se emplea cuando hay una clara diferencia entre los objetos a extraer respecto del fondo de la escena. Los principios que rigen son la similitud entre los pixeles pertenecientes a un objeto y sus diferencias respecto al resto. Por tanto, la escena debe caracterizarse por un fondo uniforme y por objetos parecidos.

Al aplicar un umbral  $T$ , la imagen en escala de grises,  $f(x,y)$ , quedará binarizada; etiquetando con '1' los pixeles correspondiente al objeto y con '0' aquellos que son del fondo. Si los objetos son claros con respecto al fondo se aplicara:

$$g(x, y) = \begin{cases} 1 & \leftrightarrow f(x, y) > T \\ 0 & \leftrightarrow f(x, y) \leq T \end{cases} \quad \text{Ec. 8.9}$$

En el caso de que los objetos sean oscuros respecto del fondo, la asignación seria a la inversa:

$$g(x, y) = \begin{cases} 1 & \leftrightarrow f(x, y) > T \\ 0 & \leftrightarrow f(x, y) \leq T \end{cases} \quad \text{Ec. 8.10}$$

El umbral puede depender de  $f(x,y)$ , de alguna propiedad local del pixel,  $p(x,y)$ , y hasta de su propia posición:

$$T = T(f(x, y), p(x, y), x, y) \quad \text{Ec. 8.11}$$

Una imagen es una función bidimensional de la intensidad del nivel de gris y contiene  $N$  pixeles cuyos niveles de gris se encuentran entre 1 y  $L$ . El número de pixeles con nivel de gris  $i$  se denota como  $f_i$ , y la probabilidad de ocurrencia del nivel de gris  $i$  en la imagen está dada por:

$$P_i = \frac{f_i}{N} \quad \text{Ec. 8.12}$$

En el caso de la umbralización en dos niveles de una imagen (a veces llamada Binarización), los píxeles son divididos en dos clases: **C1**, con niveles de gris  $[1, \dots, t]$ ; y **C2**, con niveles de gris  $[t+1, \dots, L]$ . Entonces, la distribución de probabilidad de los niveles de gris para las dos clases que son:

$$\begin{aligned} C_1: & \frac{P_1}{\omega_1(t)}, \dots, \frac{P_t}{\omega_1(t)} \\ C_2: & \frac{P_{t+1}}{\omega_2(t)}, \frac{P_{t+2}}{\omega_2(t)}, \dots, \frac{P_L}{\omega_2(t)} \end{aligned} \quad \text{Ec. 8.13}$$

Donde

$$\omega_1(t) = \sum_{i=1}^t P_i \quad \omega_2(t) = \sum_{i=t+1}^L P_i \quad \text{Ec. 8.14}$$

También, la media para la clase C1 y la clase C2 es:

$$u_1 = \sum_{i=1}^t \frac{i \cdot p_i}{\omega_1(t)} \quad u_2 = \sum_{i=t+1}^L \frac{i \cdot p_i}{\omega_2(t)} \quad \text{Ec. 8.15}$$

Sea  $\mu_T$  la intensidad media de toda la imagen. Es fácil demostrar que

$$\omega_1 \cdot \mu_1 + \omega_2 \cdot \mu_2 = \mu_T \quad \omega_1 + \omega_2 = 1 \quad \text{Ec. 8.16}$$

### 8.2.7 Eliminación de píxeles.

Se quita de una imagen binaria todos los componentes conectados (objetos) que tienen menos de P píxeles, produciendo otra imagen binaria.

### 8.2.8 Operación morfológica.

Es una técnica de procesamiento no lineal de la señal, caracterizada en realzar la geometría y forma de los objetos. Su fundamento matemático se basa en la teoría de conjunto.

#### 8.2.8.1. Dilatación.

La dilatación es una operación morfológica correspondiente a la idea intuitiva del crecimiento, o de añadir una capa de píxeles a la estructura de la imagen. La manera específica en la que es controlado el crecimiento, es por medio de una figura referida como estructura de referencia. Esta operación es definida en la notación de conjuntos de puntos como:

$$I \oplus H = \{(x', y') = (x + i, y + j) | (x', y') \in P_1, (i, j) \in P_H\} \quad \text{Ec. 8.17}$$

Como indica la expresión 9.5, el conjunto de puntos que constituyen a la dilatación de una imagen  $I$  y a la estructura de referencia  $H$ , queda definido por todas las posibles combinaciones de los pares de coordenadas de los conjuntos de puntos  $P_1$  y  $P_H$ . Se podría también interpretar a la operación de dilatación, como el resultado de añadir a los píxeles 1 de la imagen ( $P_1$ ) la forma correspondiente a la estructura de referencia.

#### 8.2.8.2. Apertura.

La apertura es definida como una erosión seguida por una dilatación, utilizando para ello la misma estructura de referencia para ambas operaciones. Lo anterior es definido como:

$$I \circ H = (I \ominus H) \oplus H \quad \text{Ec. 8.18}$$

Una apertura produce, en su primera parte (erosión), que todos los píxeles de la imagen en uno, que sean más pequeños que la estructura de referencia, sean eliminados. Las estructuras

que permanezcan en la imagen serán, a través de la dilatación, suavizadas y aumentadas aproximadamente igual a su tamaño original (antes de aplicar erosión).

### 8.2.8.3.Cierre.

A la secuencia de operaciones de dilatación seguida de la erosión, usando para ello la misma estructura de referencia, se denomina cierre. Dicha operación es definida como:

$$I^{\circ}H = (I \oplus H) \ominus H \quad \text{Ec. 8.19}$$

A través de la operación de cierre, se rellenan los hoyos encontrados dentro de los objetos de la imagen y que además son más pequeños que la estructura de referencia.

### 8.2.9. Características de objetos.

**Función bwlabel.** Permite etiquetar los objetos contenidos en una imagen binaria de acuerdo a algún criterio de vecindad. La sintaxis general de esta función es:

$$L = bwlabel(Bw, n)$$

Esta función retorna como resultado una matriz L a partir de la imagen binaria BW. Las dimensiones de L son iguales a las de BW. La matriz L contiene las etiquetas de los objetos cuyos píxeles se encuentran conectados de acuerdo a un criterio de vecindad. El parámetro n define el criterio de vecindad y solo puede adquirir dos valores 4, para el caso de 4-vecinos y 8 para el caso de 8-vecinos. El valor por defecto de n es 8 en caso de que solo se defina como argumento la función de la imagen BW.

La forma en que trabaja esta función puede ser ejemplificada de la siguiente manera:

Se define en MatLab la matriz BW:

```

BW =
1 1 1 0 1 1 1
0 1 1 0 1 1 0
1 1 1 0 0 1 0
0 0 0 1 1 0 0

```

Y se utiliza la función

```
L = bwlabel(Bw,4)
```

La matriz resultante L sería:

```

L =
1 1 1 0 1 1 1
0 1 1 0 1 1 0
1 1 1 0 0 1 0
0 0 0 1 1 0 0

```

Como puede observarse de estos resultados en tanto se utiliza la vecindad 4-vecinos, el algoritmo detecta 3 objetos diferentes, mientras que si se utiliza 8-vecinos el algoritmo reconoce solo 1 objeto.

**Función regionprops.** Permite medir un conjunto de propiedades definidas en la tabla 10.1 para cada uno de los objetos etiquetados. La sintaxis general de esta función es:

```
STATS = regionprops(L,propiedades)
```

Donde L es la matriz con los objetos etiquetados, mientras que en propiedades debe de establecerse la propiedad numérica definida en la tabla 10.1 que se desea calcular. La función regionprops regresa como resultado un arreglo de estructuras llamado STATS. Una estructura es una agrupación de datos de tipo diferente bajo un mismo nombre. Estos datos se llaman campos (fields) y son accedidos mediante el formato:

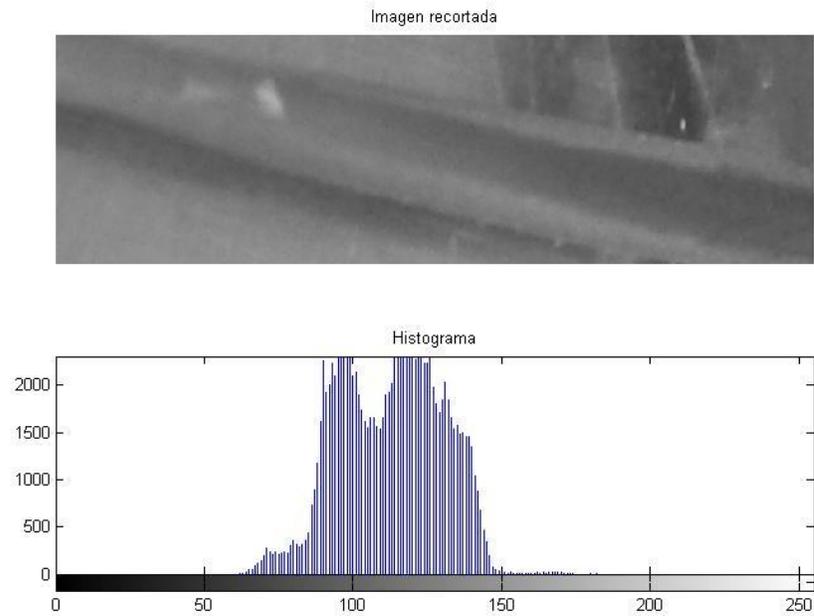
```
Dato = Nombre_de_la_estructura .field(1);
```

Por lo que el ejemplo anterior, a la variable Dato se le asigna el valor del índice 1 del campo field de la estructura Nombre\_de\_la\_estructura.

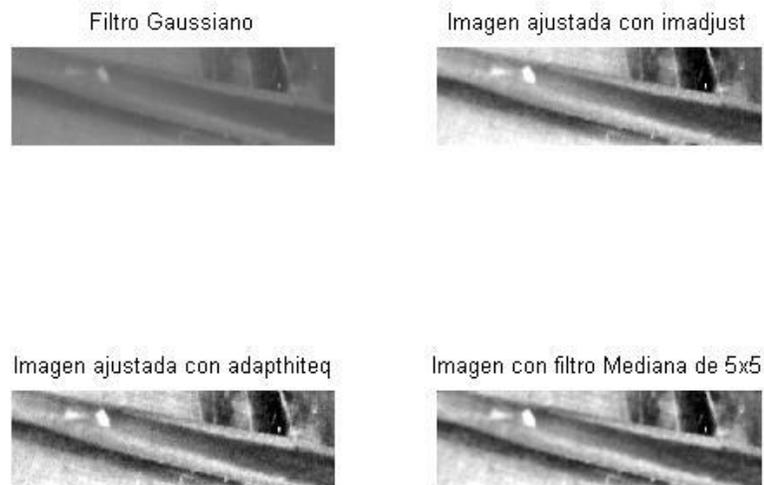
Tabla con propiedades que pueden ser calculadas por la función regionprops.

<b>Propiedad</b>	<b>Descripción</b>
<b>‘Área’</b>	Encuentra el área del objeto etiquetado
<b>‘BoundingBox’</b>	Encuentra el cuadrado mínimo que engloba al objeto
<b>‘Centroid’</b>	Encuentra al centroide del objeto
<b>‘ConvexHull’</b>	Encuentra el polígono mínimo que encierra al objeto
<b>‘ConvexArea’</b>	Encuentra el área que encierra el polígono determinado por ‘ConvexHull’
<b>‘EulerNumber’</b>	Encuentra el número de Euler del objeto
<b>‘MajorAxisLength’</b>	Encuentra el tamaño del eje que longitudinalmente es más grande en el objeto
<b>‘MinorAxisLength’</b>	Encuentra el tamaño del eje que longitudinalmente es más pequeño en el objeto
<b>‘Orientation’</b>	Encuentra la orientación del eje longitudinal máximo con respecto a la horizontal
<b>‘Perimeter’</b>	Calcula el perímetro del objeto etiquetado

### 8.3. Pruebas reales del algoritmo



*Fig. 8.3. Imagen obtenida e histograma de la misma.*



*Fig. 8.4. Aplicación de filtros*

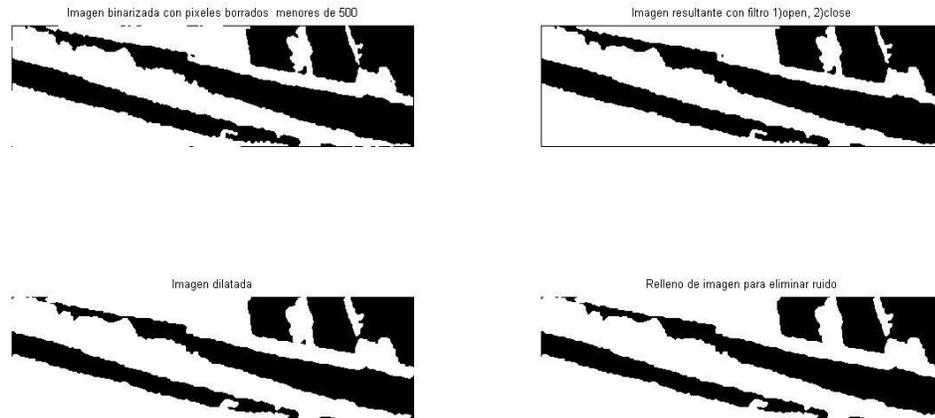


Fig. 8.5. Binarización de imagen y aplicación de operaciones morfológicas.

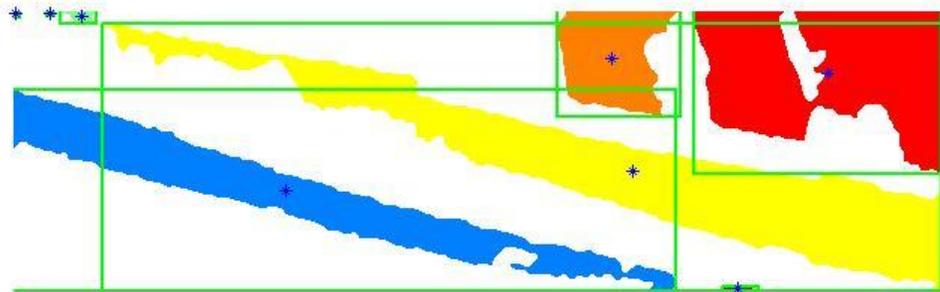


Fig. 8.6. Etiquetado de objetos encontrados.

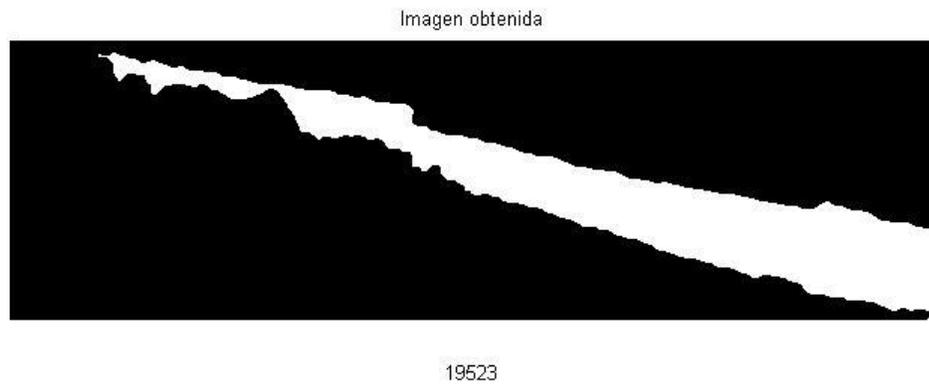


Fig. 8.7. Resultado final de la muestra obtenida en pixeles del área en blanco.

#### 8.4. Interfaz de usuario

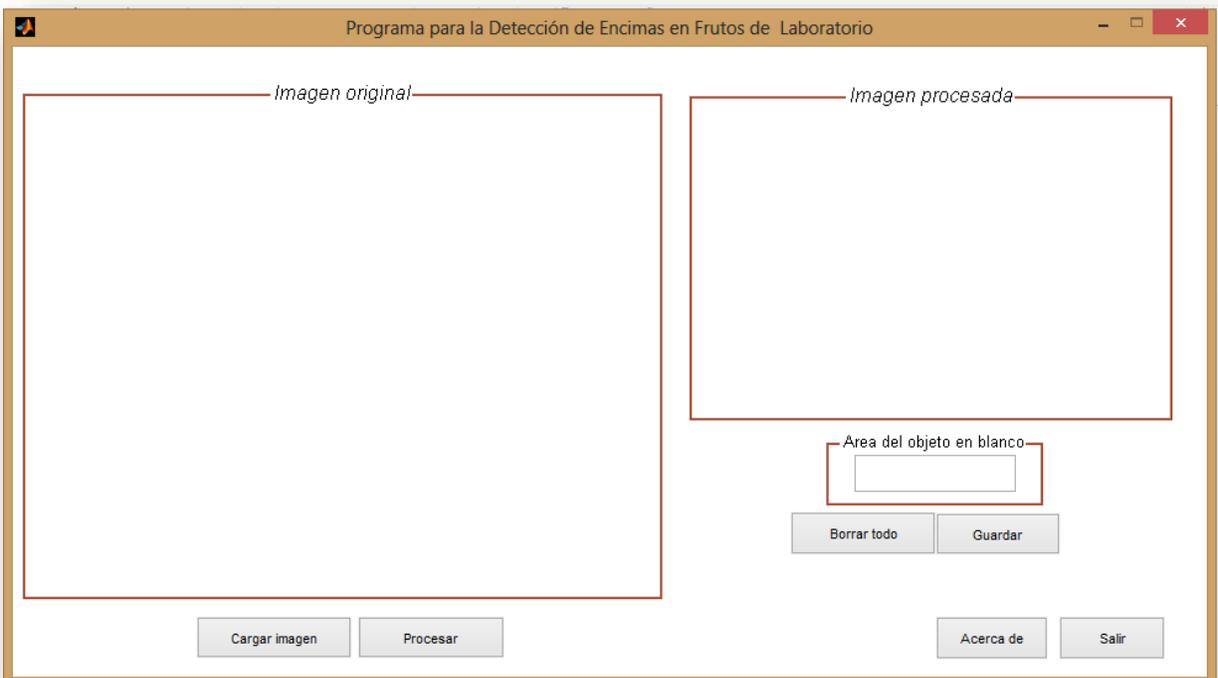


Fig. 8.8. Pantalla principal de la interfaz de usuario.

El programa fue diseñado con la herramienta Guide de MatLab, las funciones que contiene son:

- **Cargar imagen.** Abre una nueva ventana la cual nos permite elegir la imagen.
- **Procesar.** Realiza el procesamiento de la imagen cargada en el panel “Imagen original” para luego mostrar los resultados en el panel “Imagen procesada” y en el cuadro de texto ubicado en el panel área del objeto en blanco.
- **Borrar todo.** Borra la imagen la cual fue cargada el panel “Imagen original”, “Imagen Procesada” y el cuadro de texto donde es desplegado el área del objeto en blanco.
- **Guardar.** Opción para guardar los resultados obtenidos tanto la “Imagen procesada” como el área del objeto en blanco.
- **Salir.** Cierre del programa.
- **Acerca de.** Despliega un cuadro de dialogo donde muestra cuando fue diseñado el programa, el autor del mismo, y el nombre del proyecto asignado.

### 8.5. Resultados del Algoritmo

Mediante el algoritmo elaborado en MatLab y los filtros aplicados para el procesamiento de imágenes hicieron que los resultados fueran satisfactorios, la disminución de ruido fue considerable sin embargo para resultados óptimos habrán que hacer ciertas consideraciones.

## 9. Conclusiones

Mediante la herramienta de software MatLab el procesamiento de imágenes digitales se facilitó debido a que las imágenes se manejan de manera matricial en lo cual dicho software se especializa. Aplicando diferentes filtros para el tratamiento de imágenes dieron como resultado los mencionados en el punto anterior.

El procesamiento de imágenes se facilitó debido a las opciones que tiene la interfaz de usuario la cual fue diseñada con la herramienta Guide de MatLab. Algunas de las ventajas que

conllevar usar dicha interfaz es la opción de elegir la imagen a procesar, de lo cual si no es la que se desea procesar podremos borrarla, la principal función que contiene es la de guardar el resultado obtenido después de haber procesado la imagen.

## **10. Recomendaciones**

En el procesamiento de imágenes digitales se requiere la eliminación de ruido o reducción del mismo, sin embargo la imagen a procesar pudo no obtenerse de manera óptima lo cual nos lleva a tener medidas a considerar en la captura de imágenes tales como:

- Luz
- Área de trabajo específico para la adquisición de imágenes.
- Evitar colocar objetos que no sean necesarios en el área de trabajo.
- Evitar superficies reflejantes.
- Aplicar filtros a la cámara para obtener mejor calidad en la adquisición de imágenes.

Mantener un ambiente controlado hará que el ruido sea menor en la adquisición de imágenes lo cual facilitará el procesamiento de las mismas y así obtener mejores resultados.

## **11. Referencias**

[1] Erik Cuevas - Daniel Zaldívar - Marco Pérez, Procesamiento digital de imágenes con MATLAB y Simulink, Alfaomega, Ra-ma, Primera edición, 2010.

[2] Rastislav Lukac, Konstantinos N. Plataniotis, Color Image Processing Methods and Applications, CRC Taylor & Francis, 2007.

[3] Rafael C- González, Richard E. Woods, Digital Image Processing, Prentice Hall, Segunda edición, 2002.

[4] Maïtine Bergounioux, Mathematical Image Processing, Springer, 2011.

[5] John C. Russ, The Image Processing Handbook, Springer, IEEE Press, CRC PRESS, Tercera edición, 1998.

Anexos

**SONY**

4-410-736-23(1)

Digital Still Camera / Instruction Manual	GB	
Cámara fotografía digital / Manual de instrucciones	ES	

**Cyber-shot**



DSC-W630/W650

## Especlflcaciones

### Cámara

#### [Sistema]

Dispositivo de imagen: CCD de color de 7,75 mm (tipo 1/2,3), filtro de color primario

Número total de píxeles de la cámara: aprox. 16,4 megapíxeles

Número efectivo de píxeles de la cámara: aprox. 16,1 megapíxeles

Objetivo: objetivo zoom Carl Zeiss Vario-Tessar de 5×  
f = 4,5 mm – 22,5 mm (25 mm – 125 mm (equivalente a una película de 35 mm))  
F2,6 (W) – F6,3 (T)

Durante la filmación de películas (16:9): 29 mm – 145 mm\*

Durante la filmación de películas (4:3): 35 mm – 175 mm\*

\* Cuando [SteadyShot película] está ajustado en [Estándar].

SteadyShot: óptico

Control de exposición: exposición automática, selección escena (11 modos)

Balance de blancos: Automático, Luz diurna, Nublado, Fluorescente 1/2/3, Incandescente, Flash, Pulsación única

Intervalo de grabación en el modo de Ráfaga: aprox. 1,0 s (segundo)

Formato de archivo:

Imágenes fijas: compatible con JPEG (DCF, Exif, línea de base MPF), compatible con DPOF

Películas: MPEG-4 AVC/H.264

Soporte de grabación: memoria interna (aprox. 27 MB) "Memory Stick Duo", tarjetas SD

Flash: alcance del flash (sensibilidad ISO (Índice de exposición recomendado) ajustada en Automático):

Aprox. de 0,3 m a 3,7 m (W)

Aprox. de 1,0 m a 1,5 m (T)

#### [Conectores de entrada y salida]

Toma USB / A/V OUT:

Salida de vídeo

Salida de audio

Comunicación USB

Comunicación USB: Hi-Speed USB (USB 2.0)

#### [Pantalla]

Pantalla LCD:

DSC-W630:

Unidad TFT de 6,7 cm (tipo 2,7)

DSC-W650:

Unidad TFT de 7,5 cm (tipo 3,0)

Número total de puntos:

230 400 puntos

#### [Alimentación, general]

Alimentación: batería recargable

NP-BN, cc 3,6 V

Consumo de energía (durante la toma de imágenes): 1,1 W

Temperatura de funcionamiento:

De 0 °C a 40 °C

Temperatura de almacenamiento:

De -20 °C a +60 °C

Dimensiones (cumple con los estándares de la CIPA):

DSC-W630: 91,0 mm × 52,2 mm × 19,1 mm (an/al/prf)

DSC-W650: 94,0 mm × 56,0 mm × 19,1 mm (an/al/prf)

ES

ES  
23

Peso (cumple con los estándares de la CIPA) (incluida la batería NP-BN y el "Memory Stick Duo"):

DSC-W630:

Aprox. 116 g

DSC-W650:

Aprox. 124 g

Micrófono: monoaural

Altavoz: monoaural

Exif Print: compatible

PRINT Image Matching III:  
compatible

### **Cargador de batería BC-CSN/ BC-CSNB**

Requisitos de alimentación: ca 100 V a  
240 V 50 Hz/60 Hz 2 W

Tensión de salida: cc de 4,2 V 0,25 A

Temperatura de funcionamiento:

De 0 °C a 40 °C

Temperatura de almacenamiento:

De -20 °C a +60 °C

Dimensiones:

Aprox. 55 mm × 24 mm × 83 mm  
(an/al/prf)

Peso: aprox. 55 g

### **Batería recargable NP-BN**

Batería utilizada: batería de iones de litio

Tensión máxima: cc de 4,2 V

Tensión nominal: cc de 3,6 V

Tensión de carga máxima: cc 4,2 V

Corriente de carga máxima: 0,9 A

Capacidad:

normal: 2,3 Wh (630 mAh)

mínima: 2,2 Wh (600 mAh)

El diseño y las especificaciones están sujetos a cambios sin previo aviso.

### **Marcas comerciales**

- Las siguientes marcas son marcas comerciales de Sony Corporation. **Cyber-shot**, "Cyber-shot", "Memory Stick PRO Duo", "Memory Stick PRO-HG Duo", "Memory Stick Duo"
- Windows es una marca comercial registrada de Microsoft Corporation en los Estados Unidos y/o en otros países.
- Mac es una marca comercial registrada de Apple Inc.
- El logotipo de SDXC es una marca comercial de SD-3C, LLC.
- Además, los nombres de sistemas y productos utilizados en este manual son, en general, marcas comerciales o marcas comerciales registradas de sus respectivos programadores o fabricantes. Sin embargo, las marcas <sup>TM</sup> o © no se utilizan en todos los casos en este manual.

En la página Web de atención al cliente puede encontrar información adicional sobre este producto y respuesta a las preguntas hechas con más frecuencia.

<http://www.sony.net/>

Impreso en papel reciclado en un 70% o más utilizando tinta hecha con aceite vegetal exento de compuesto orgánico volátil (COV).