

Actividad de Semana i

Taller de Procesamiento de Imágenes Utilizando Matlab y Visita a CICATA-IPN Unidad Qro

M. en C. Elizabeth Chávez Hernández

26 de Septiembre de 2017

Elizabeth Chávez Hernández

Licenciatura en Matemáticas Aplicadas.
Facultad de Ciencias Físico Matemáticas, UAdeC.

Maestría en Ciencias de la Computación.
Instituto de Investigaciones en Matemáticas Aplicadas
y en Sistemas (IIMAS-UNAM).

Docente en: Facultad de Ingeniería (UAdeC), Facultad
de Ingeniería (UACH), Facultad de Ciencias (UNAM),
Universidad Politécnica de Querétaro,
ITESM-Querétaro.

email: echavezhe@itesm.mx

Contenido

2. Conceptos básicos de Imágenes

2.1. Representación de Matrices

2.2. Operaciones con Matrices

2.3. Visualizar Matrices

2.5. Leer y guardar imágenes en Matlab.

2.6. Mostrar imágenes en Matlab y secciones de ventanas

2.7. Acceso a áreas de interés en la imagen

2.8. Manipular imágenes en Matlab

2.1. Representación de Matrices

En MATLAB una imagen a escala de grises es representada por medio de una matriz bidimensional de $m \times n$ elementos en donde n representa el numero de pixeles de ancho y m el numero de pixeles de largo.

Representación Matemática de una Matriz

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}$$



2.1. Representación de Matrices

- ✓ Las imágenes binarias sólo tienen un plano y contienen únicamente dos valores: 0 que representa al negro y 1 que representa al blanco.

Hacer los siguientes ejercicios:

Crear una imagen toda negra

```
A=zeros(m,n);
```

Crear una imagen toda blanca

```
B=ones(m,n);
```

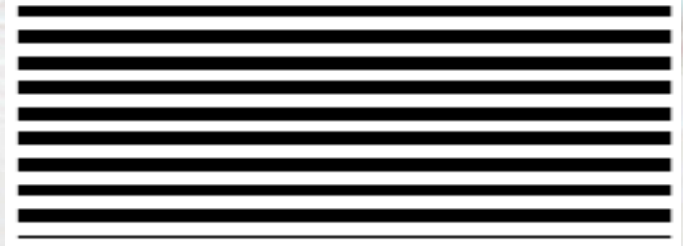
Crear dos imágenes con patrones diferentes

```
C=zeros(192,512);
```

```
C(1:96,1:256)=ones(96,256);
```

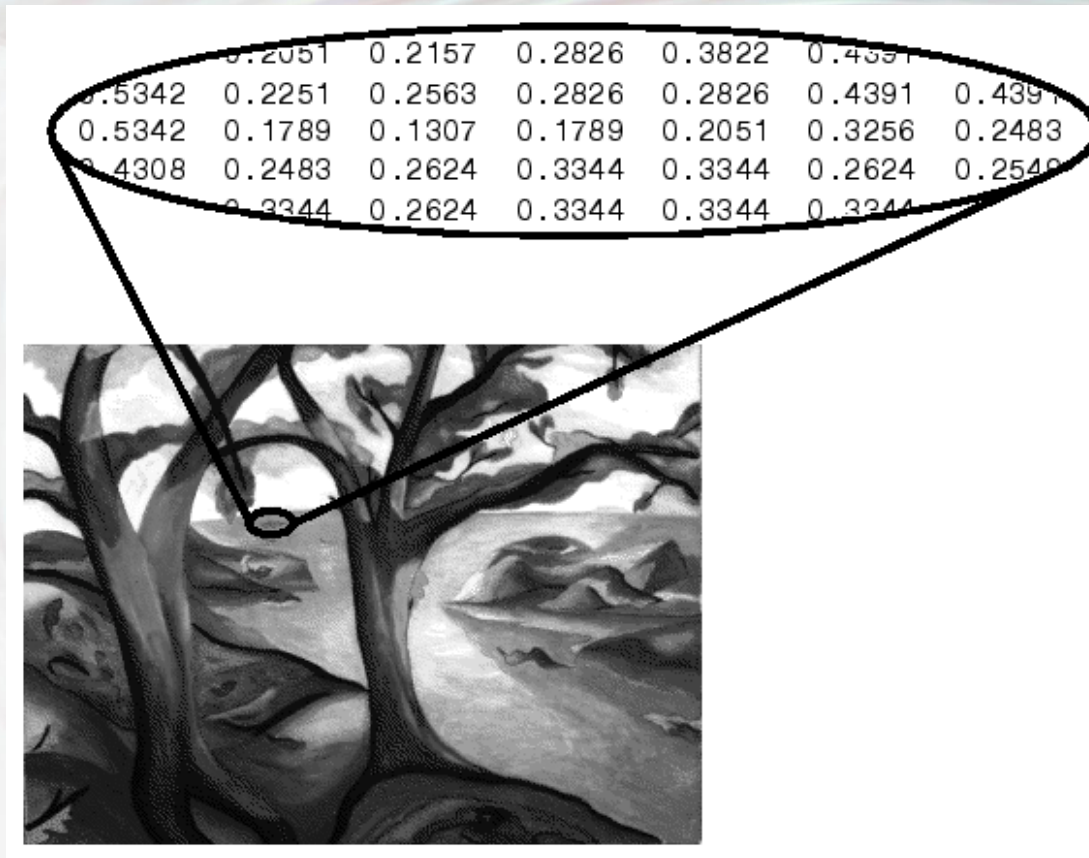
```
C(97:192,257:512)=ones(96,256);
```

Franjas horizontales



2.1. Representación de Matrices

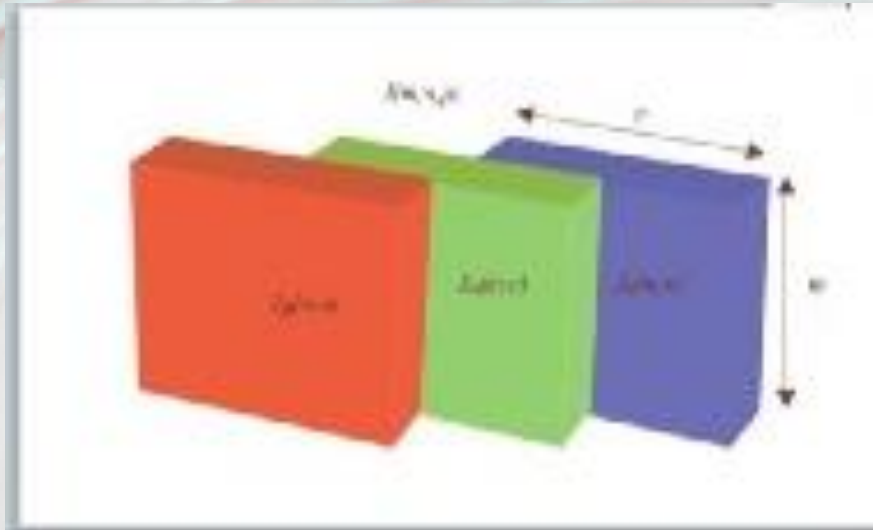
- ✓ Las imágenes a escala de grises sólo tienen un plano, constituido por la matriz $m \times n$ que contiene los valores de intensidad para cada índice (pixel).



Las matrices pueden ser de tipo:

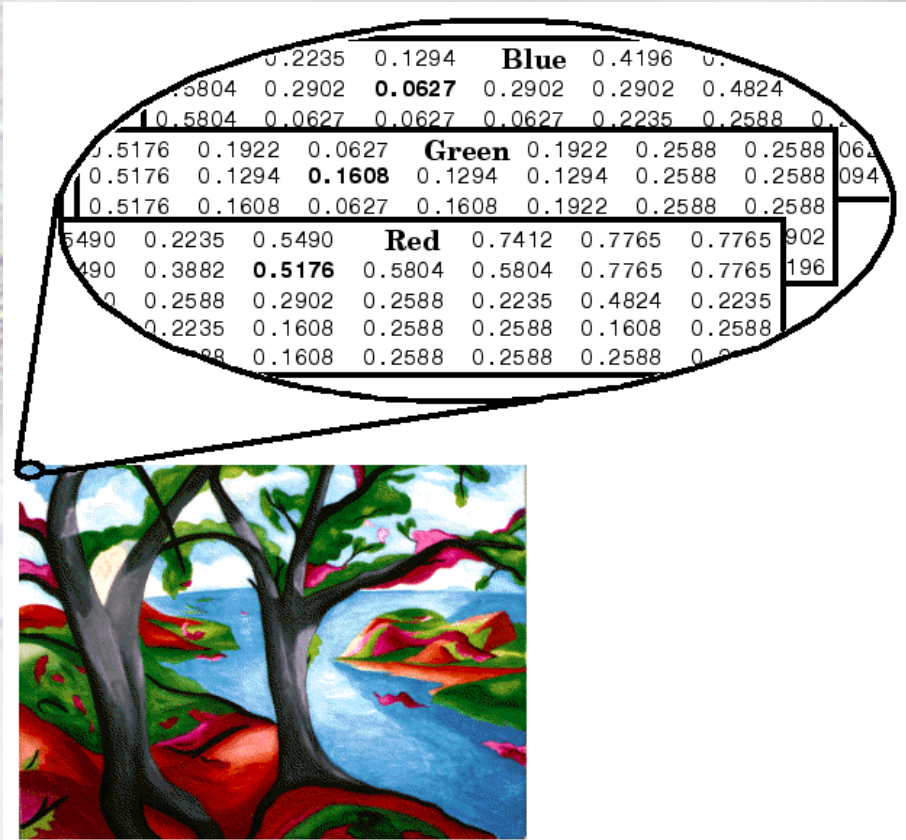
- doble (0 – 1)
- uint8 (0 - 255)
- uint16 (0 - 65,535)

- ✓ Las imágenes de color cuentan con mas de un plano. Las imágenes RGB se representan con una matriz tridimensional $m \times n \times p$, donde p representa el plano. Los 3 planos (o capas) son para representar el color, 1 para el rojo, 2 para el verde y 3 para el azul.



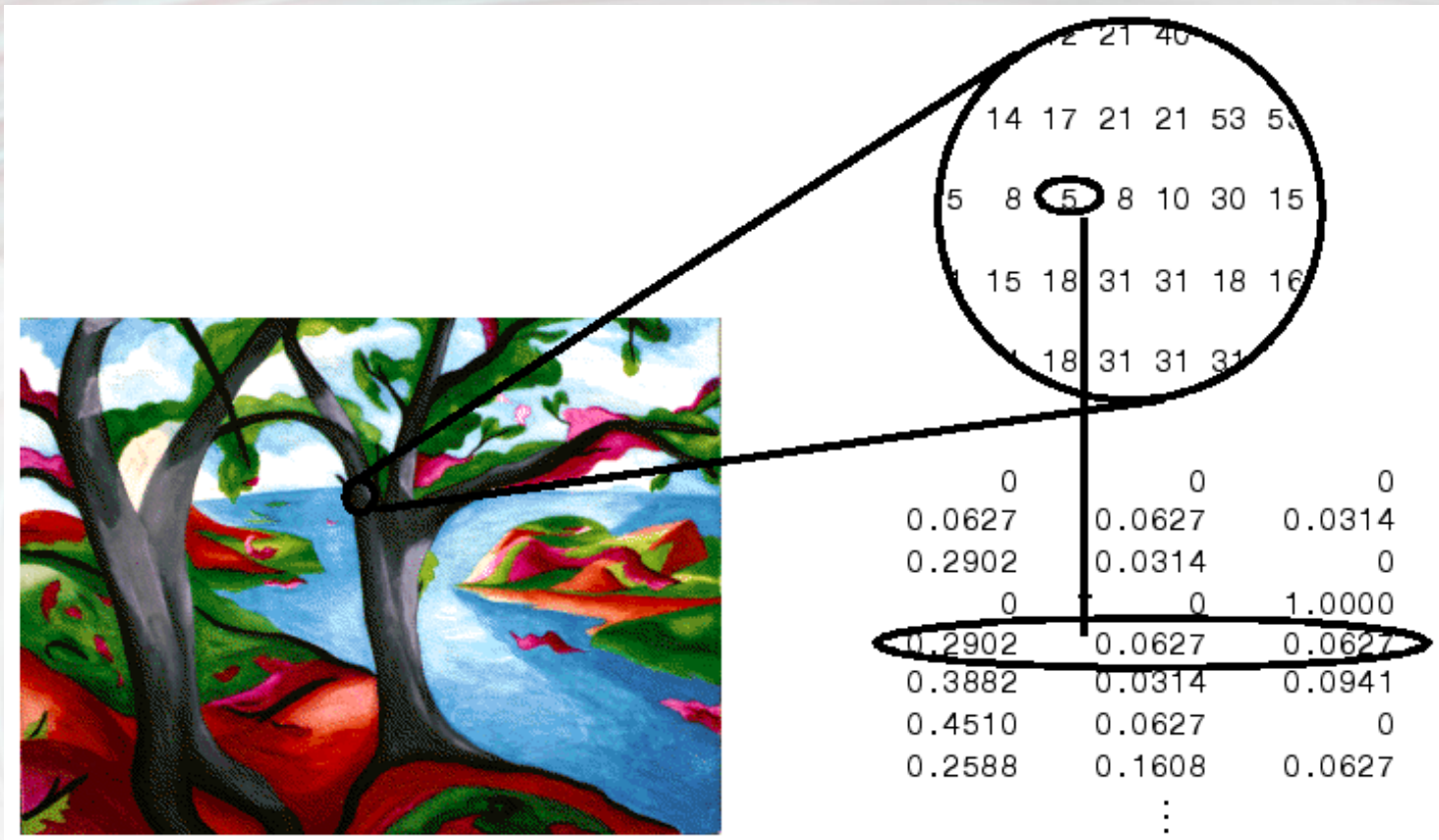
Obtener cada uno de los planos que la componen:

```
planeR=image2( :, :,1) ;
planeG=image2( :, :,2) ;
planeB=image2( :, :,3) ;
```



Una imagen RGB de tipo uint8 nos da lugar a 256 tonos de cada color (R, G y B), lo que nos da un total de 16 millones de colores.

- ✓ Las imágenes indexadas, al igual que las de escala de grises, tienen sólo un plano. La diferencia es que los números que contiene *no representan un tono de gris*, sino la posición del color del pixel dado en un mapa de color.



El mapa de color es un arreglo que puede tener tantas filas como se desee pero debe contener tres columnas, cada fila representa un color dado por la combinación de RGB guardada en las tres columnas.

2.1. Representación de Matrices

Comandos de conversión de imágenes en MATLAB

Comando	Descripción
gray2ind	Crea una imagen indexada a partir de una imagen en escala de grises.
im2bw	Crea una imagen binaria a partir de una imagen indexada o RGB.
ind2rgb	Crea una imagen RGB a partir de una imagen indexada.
rgb2gray	Crea una imagen en escala de grises a partir de una imagen RGB.
rgb2ind	Crea una imagen indexada a partir de una imagen RGB.

2.1. Representación de Matrices

Formatos y extensiones soportadas por MATLAB

Formato	Extensión
TIFF	.tiff
JPEG	.jpg
GIF	.gif
BMP	.bmp
PNG	.png
XWD	.xwd

2.2. Operaciones con Matrices

En MATLAB podemos trabajar operaciones matriciales de forma directa.

Operación	Significado
A+B	Suma de Matrices ✓ Misma dimensión de Matrices
A-B	Diferencia de Matrices ✓ Misma dimensión de Matrices
A.*B	Producto de Matrices componente a componente. ✓ Misma dimensión de Matrices
A./B	División de Matrices componente a componente. ✓ Cuando Matriz A tiene la misma cantidad de filas que B.
A.^2	Cuadrado componente a componente
A*B	Multiplicación de matrices. Matriz A debe tener mismas columnas que renglones de matriz B.

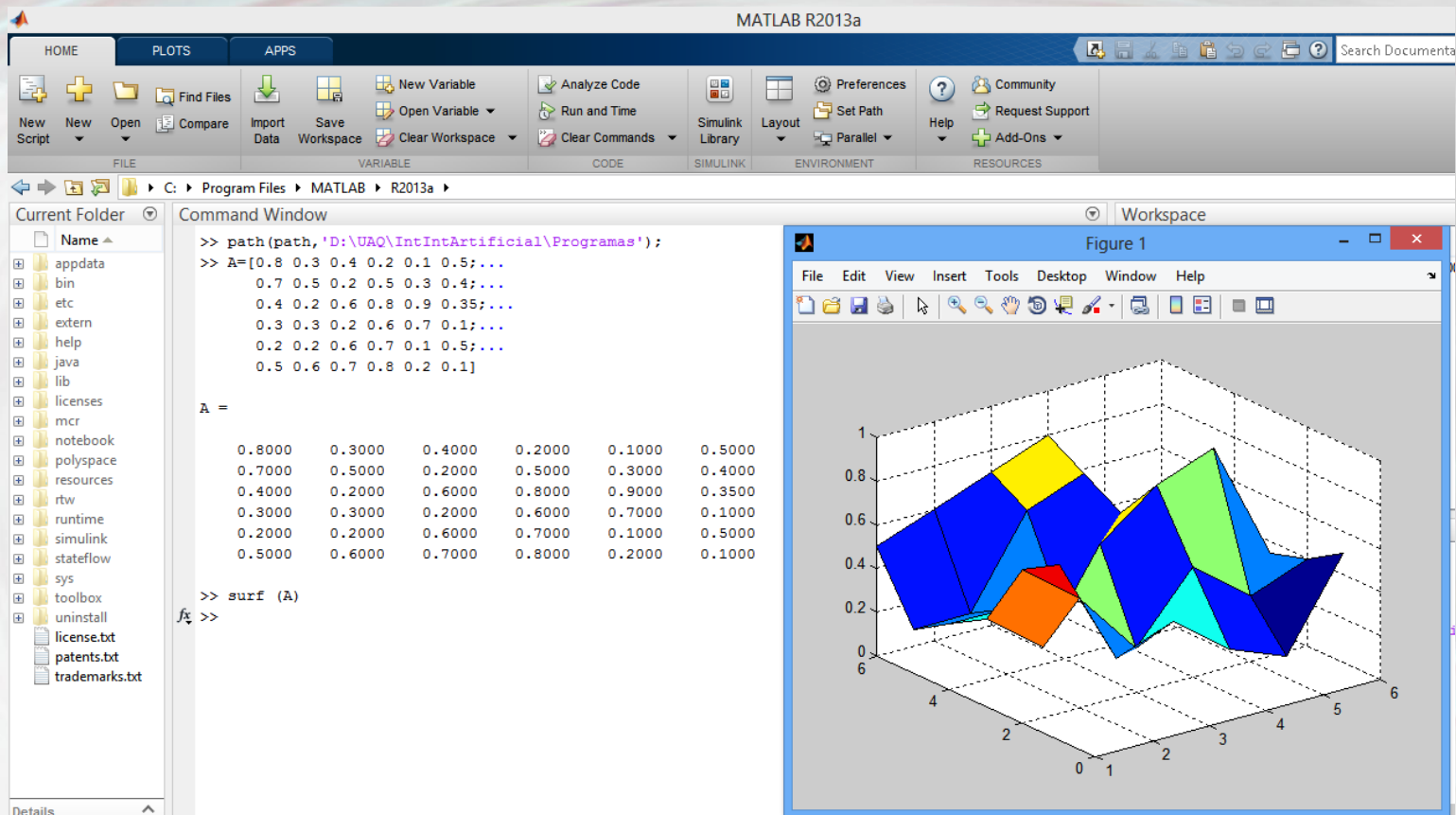
2.2. Operaciones con Matrices

En MATLAB podemos utilizar funciones que nos sirven para generar diferentes matrices.

Operación en Matlab	Significado
A'	Calcular la matriz transpuesta.
size(A)	Calcula el tamaño de una Matriz. Ren-Col.
inv(A)	Calcula la inversa de una Matriz.
eye(3)	Genera la matriz identidad cuadrada.
ones(3,2)	Llena una matriz de 3x2 de unos.
zeros(3,2)	Llena una matriz de 3x2 de ceros.
rand(3,2)	Para generar una matriz con números aleatorios uniformemente distribuidos entre 0 y 1.
magic(3)	Genera una matriz cuadrada tal que la suma de los elementos de cada fila son iguales, también lo es la suma de los elementos de cada columna.

2.3. Visualizar Matrices

- `surf(matriz);`



2.4. Secciones de ventanas

`subplot (num.renglones, num.columnas, ubicación);`

La ubicación se numera de arriba abajo de izquierda a derecha.

```
figure;
```

```
subplot(2,2,1);
```

```
imshow(I);
```

```
title('Imagen original');
```

```
subplot(2,2,2);
```

```
mshow(IND);
```

```
title('Indexada');
```

```
subplot(2,2,3);
```

```
imagesc(G);
```

```
title('Escala de grises');
```

```
colormap(gray);
```

```
subplot(2,2,4);
```

```
imshow(TC(:,:,2));
```

```
title('RGB (capa Verde)');
```


2.5. Leer y guardar imágenes en Matlab

- Leer imagen

```
lori=imread('c:\ara\celula.jpg');
```

- Convertir RGB a Gris

```
I=rgb2gray(lori);
```

- Guardar una matriz como imagen

```
imwrite(I2, 'c:\ara\celula2.jpg');
```

2.5. Leer y guardar imágenes en Matlab

Ejercicio 1 en Clase

Caso 1. A partir de una imagen RGB, obtener una imagen en escala de grises.

Caso 2. A partir de una imagen en escala de grises, obtener una imagen binaria.

Visualizar en Matlab las 3 ventanas de: imagen original, escala de gris y binaria.

2.6. Manipular imágenes en Matlab

- Para acceder a un área específica de la imagen debemos indicar, en la capa deseada, desde y hasta qué fila deseamos ver y desde y hasta que columna deseamos ver.

Ejemplo:

```
I=imread('peppers.png');  
imshow(210:310,210:360,2)
```

Se puede, si se desea, cambiar el valor de un pixel:

$I(100,100) = 0$

2.6. Manipular imágenes en Matlab

- Leer imagen

```
lori=imread('celula.jpg');
```

- Convertir RGB a Gris

```
I=rgb2gray(lori);
```

- Mostrar una imagen en pantalla.

```
imshow(I);
```

2.6. Manipular imágenes en Matlab

- Redimensiona el tamaño de la imagen

```
redim=imresize(I, [50 50]);
```

- Rotar una imagen

```
rot=imrotate(I,90);
```

```
imshow(rot)
```


2.6. Manipular imágenes en Matlab

Ejercicio 2 en Clase

Caso 1. A partir de una imagen RGB, obtener una imagen en escala de grises.

Caso 2. A partir de una imagen en escala de grises, Hacer una rotación.

Caso 3. A partir de una imagen en escala de grises, Hacer una redimensión a la imagen.

Visualizar en Matlab las 4 ventanas de: imagen original, escala de gris, rotada y redimensionada.

2.7. Dilatación y Erosión de una imagen

Son operaciones realizadas sobre imágenes binarias basadas en formas. La dilatación adiciona píxeles en las fronteras de los objetos.

```
lori=imread('celula.jpg');
```

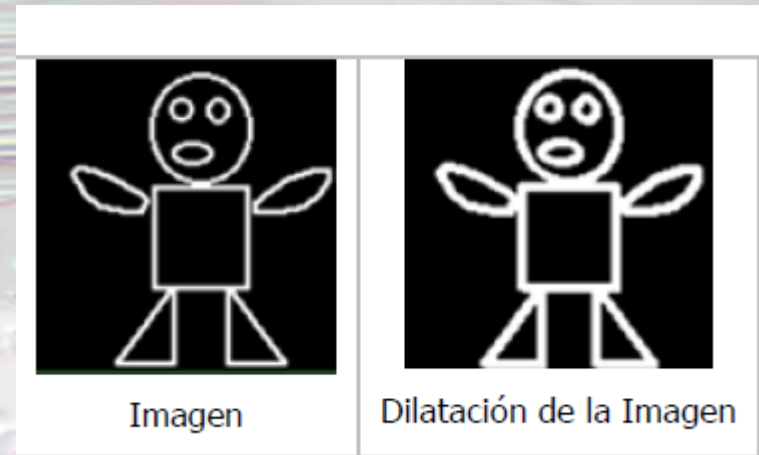
```
lb=im2bw(lori);
```

- Dilatación de una imagen

```
w=eye(3)
```

```
ldil=imdilate(lb,w);
```

```
imshow(ldil)
```



2.7. Dilatación y Erosión de una imagen

1.- Leer imagen

```
bw = imread('text.png');
```

2.- Crea un elemento de estructura morfológica con una variedad de formas y tamaños.

```
se = strel('line',11,90);
```

3.- Realiza la dilatación de la imagen

```
bw2 = imdilate(bw,se);
```

4.- Muestra ambas imágenes.

```
imshow(bw), title('Imagen Original')  
figure, imshow(bw2), title('Imagen Dilatada')
```


2.7. Dilatación y Erosión de una imagen

Son operaciones realizadas sobre imágenes binarias basadas en formas. La erosión remueve píxeles en las fronteras de los objetos.

```
lori=imread('c:\ara\celula.jpg');  
lb=im2bw(lori);
```

- Erosión de una imagen

```
w=eye(3)  
lero=imerode(lb,w);  
imshow(lero)
```

2.7. Dilatación y Erosión de una imagen

Ejemplo.- El objetivo es hacer erosión en una imagen binaria con un elemento estructurante disco.

1.- Leer la imagen

```
Imorig = imread('circles.png');
```

2.- Eliminar los círculos que tienen un radio de menos de 11 píxeles abriéndolo con el elemento de estructuración en forma de disco.

```
se = strel('disk',11);
```

3.- Realiza la erosión en la imagen

```
erodedBW = imerode(Imorig,se);
```

4.- Muestra ambas imágenes.

```
imshow(Imorig), figure, imshow(erodedBW)
```


2.7. Dilatación y Erosión de una imagen

Ejercicio 3 en Clase

1.- Hacer un programa con una imagen de Matlab cameraman.tif. El objetivo será hacer dilatación en la imagen.

2.- Hacer un programa con una imagen de Matlab circbw.tif. El objetivo será hacer erosión en la imagen.

Visualizar en Matlab las 4 ventanas de: imagen original 1, Original 2, imagen dilatada y erosión.

Bibliografía

- 1.- Etter M. Delores. **Solución de Problemas de Ingeniería con Matlab**. 2ª Ed. Prentice Hall. 1997.
- 2.- Javier García de Jalón *et al.* **Aprenda Matlab 7.0 Como si estuviera en primero**. Universidad Politécnica de Madrid. 2005.
- 3.-
http://www.elai.upm.es/webantigua/spain/Asignaturas/MIP_VisionArtificial/PracticasVA/prac2VA_AdquisicionGUI.pdf