

1 Configuración de entorno

Sentencia	Ejemplo
<code>clc</code>	Borrar comandos en consola.
<code>clear all</code>	Limpiar valores en memoria.
<code>close all</code>	Cerrar ventanas abiertas.

2 Entradas y salidas

Sentencia	Ejemplo
<code>fprintf</code>	<i>Impresión en pantalla con formato</i> <pre>ti=2.56; paso=0.5; va=0; vi=0; vti=0; fprintf('ti=%.2i seg ti+paso=%.2f seg va=%.2i km/seg vi= %.2f km/seg vti= %.2f km/seg \n',ti,ti+paso,va,vi,vti);</pre>
<code>disp</code>	<i>Impresión en pantalla de cualquier tipo de datos</i> <pre>Vti=0.9; disp(vti); %muestra valor solo disp('mensaje'); %muestra mensaje</pre>
<code>input</code>	<i>Ingreso por teclado</i> <pre>M=input('Mensaje entrada1->'); B=input('Mensaje entrada2 ->'); disp(M); disp(B);</pre>

3 Control del flujo y variables

Sentencia	Ejemplo
<code>if ... end</code>	<pre>ti=2; if ti==2 fprintf('Igual a dos'); else fprintf('No es igual a dos'); end ti=2; vi=0; if (ti==2 & vi==0) fprintf('ti=2 y vi=0 \n'); else fprintf('no cumple \n'); end</pre>
Existe variable <code>switch .. end</code>	<pre>valor=input('Mensaje entrada1->');</pre>

Sentencia	Ejemplo
	<pre>switch valor case 1 disp('Valor -> 1.');</pre>
	<pre>case 2 disp('Valor -> 2.');</pre>
	<pre>case 3 disp('Valor -> 3.');</pre>
	<pre>otherwise disp('Otros valores .');</pre>
	<pre>end</pre>
while .. end	<pre>Ti=0; paso=0,5; tiempoFinal=10; while ti < tiempoFinal fprintf('t= %.2f paso=%.2f \n',ti,paso); ti=ti+paso; end</pre>
for ... end	<pre>for ti=0:tiempoFinal fprintf('t= %.2f \n',ti); end</pre>

4 Funciones y procedimientos

Sentencia	Ejemplo
function ... end	<p><i>Función en archivo .m</i></p> <pre>function [salida1 salida2] = untitled2(entrada1, entrada2) fprintf('entrada1= %.2f \n entrada2= %.2f \n',entrada1, entrada2); salida1=10; salida2=20 end</pre> <p>Llamar desde la lines de comandos como [a,b]=ejemplofuncion(1,2);</p>
@	<p><i>Función en línea dentro de un script</i></p> <pre>%definición de función en línea InterpolacionLinealGrado1 = @(x0,fx0,x1,fx1,x) (fx0+((fx1-fx0)/(x1- x0))*(x-x0)); %llamado a función en línea valorGrado1=InterpolacionLinealGrado1(1 ,3,5,99,4);</pre>

5 Matrices y vectores

Sentencia		Ejemplo
X=[1:10];	<i>Definición y acceso a un vector</i>	<code>X=[1:10]; X(indice);</code>
M(fila, columna);	<i>Definición y acceso a una matriz</i>	<code>M=[1, 2, 3; 4, 5, 6; 7, 8, 9] M(fila, columna);</code>
size(X)	<i>Tamaño de una matriz</i>	<code>[filas,columnas]=size(X);</code>
zeros(n,n)	<i>Inicializar con ceros</i>	<code>matriz=zeros(n,n);</code>
	<i>Recorrer y mostrar una matriz</i>	<code>X=[1, 2, 3; 4, 5, 6; 7, 8, 9] [n,m]=size(X); for i=1:1:n for j=1:1:(m) fprintf('%.2f ',X(i,j)); end fprintf('\n'); end</code>
cat(1,A,I)	<i>Concatenar matrices</i>	<code>A=[1 2 3;4 5 6] I=[1 2 3;4 5 6] cat(2,A,I) %concatenar columnas cat(1,A,I) % concatenar filas</code>
A'	<i>Transpuesta de una matriz</i>	<code>A=[1 2 3;4 5 6] A'</code>
det(A)	<i>Determinante</i>	<code>A=[1 2 3;4 5 6; 7 8 9] determinante=det(A)</code>

6 Ploteado de funciones

Sentencia	Ejemplo
-----------	---------

7 Procesamiento digital de imágenes

Sentencia		Ejemplo
	<i>Abrir imagen</i>	<code>Iorig=imread('imagen.jpg');</code>
imshow(Iorig);	<i>Mostrar imagen</i>	<code>Iorig=imread('imagen.jpg');</code>

Sentencia		Ejemplo
	<i>Guardar imagen</i>	<pre>figure; imshow(Iorig); Iorig=imread('imagen.jpg'); imwrite(Iorig,'prueba.jpg','jpg')</pre>
	<i>Acceder a la matriz de la imagen.</i>	
size(originalImage)	<i>Tamaño de imagen, ancho y alto</i>	<pre>Iorig=imread('imagen.jpg'); [rows, columns, numberOfColorBands]=size(Iorig);</pre>
im2bw(Iorig,nivel)	<i>Binarizar imagen. 1= BLANCO. 0=NEGRO.</i>	<pre>Iorig=imread('imagen.jpg'); nivel = graythresh(Iorig); %umbral de nivel de gris IB2=im2bw(Iorig,nivel); figure;imshow(IB2); %mostrar</pre>
strel('disk', 3)	<i>Definición de elementos estructurantes</i>	<pre>SE = strel('disk', 3); %disco con radio 3 pixeles SE = strel('diamond', 5); %disco con radio 5 pixeles SE = strel('line', 3, 0); %línea 3 pixeles, 0 grados % otros: ball, octagon, square</pre>
imerode(IB2,SE)	<i>Erosión</i>	<pre>SE = strel('disk', 3); IB4 = imerode(IB2,SE); figure;imshow(IB4);</pre>
imdilate(IB2,SE)	<i>Dilatación</i>	<pre>SE = strel('disk', 3); IB4 = imdilate(IB2,SE); figure;imshow(IB4);</pre>
imclose(IB5,SE)	<i>Apertura</i>	<pre>SE = strel('disk', 3); IB7=imclose(IB5,SE);</pre>
imclose(IB5,SE)	<i>Cerradura</i>	<pre>SE = strel('disk', 3); IB7=imclose(IB5,SE);</pre>
	<i>Invertir imagen binaria</i>	<pre>Iorig=imread('imagen.jpg'); IB1=im2bw(Iorig,nivel); IB2=1-IB1; figure; imshow(IB1); figure; imshow(IB2);</pre>
bitxor(IB7,IB10)	<i>XOR de imágenes</i>	<pre>Iorig1=imread('imagen.jpg'); Iorig2=imread('imagen.jpg'); IB1=im2bw(Iorig1,nivel); IB2=im2bw(Iorig2,nivel); IB3=bitxor(IB1,IB2); figure; imshow(IB1); figure; imshow(IB2);</pre>

Sentencia	Ejemplo
	figure; imshow(IB3);
bitor (IB1,IB2) <i>OR de imágenes</i>	Iorig1=imread('imagen.jpg'); Iorig2=imread('imagen.jpg'); IB1=im2bw(Iorig1,nivel); IB2=im2bw(Iorig2,nivel); IB8=bitor(IB1,IB2); figure; imshow(IB1); figure; imshow(IB2);
rgb2gray (Iorig) <i>Convertir de RGB a grises</i>	Iorig=imread('imagen.jpg'); IG=rgb2gray(Iorig);
edge (IG,'prewitt') <i>Filtro Sobel, realce de bordes</i>	Iorig=imread('imagen.jpg'); IG=rgb2gray(Iorig); Iedges=edge(IG,'prewitt'); %puede definirse el umbral de gris figure;imshow(Iedges)
rgb2gray (Iorig) <i>Filtro Laplaciano, realce de bordes</i>	Iorig=imread('imagen.jpg'); IG=rgb2gray(Iorig); Iedges=edge(IG,'log'); %puede definirse el umbral de gris figure;imshow(Iedges)
edge (IG,'prewitt') <i>Filtro Prewitt, realce de bordes</i>	Iorig=imread('imagen.jpg'); IG=rgb2gray(Iorig); Iedges=edge(IG,'prewitt'); %puede definirse el umbral de gris figure;imshow(Iedges)
edge (IG,'canny') <i>Filtro Canny</i>	Iorig=imread('imagen.jpg'); IG=rgb2gray(Iorig); Iedges=edge(IG,'canny'); figure;imshow(Iedges)
fspecial ('average') <i>Filtro de la media</i>	Iorig=imread('imagen.jpg'); IB1 = im2bw(Iorig); f=fspecial('average'); IB2=filter2(f,IB1);
imfill (IB1,'holes') <i>Llenado de agujeros</i>	Iorig=imread('imagen.jpg'); IB1 = im2bw(Iorig); IB2 = imfill(IB1,'holes'); imshow(BW4), figure, imshow(BW5)
regionprops (IB1,'Ar <i>Cálculo de áreas</i> ea')	area=regionprops(IB1,'Area') fprintf('Area= %.2i; area.Area);
regionprops (IB1,'Pe <i>Cálculo de</i> rimeter ') <i>perímetro</i>	perimetro=regionprops(IB1,'Perimeter') fprintf('Perimetro= %.2f \n',

Sentencia		Ejemplo
		<code>perimetro.Perimeter);</code>
<code>regionprops(IB1,'Eccentricity')</code>	<i>Cálculo de excentricidad</i>	<code>excentricidad=regionprops(IB1,'Eccentricity') fprintf('Excentricidad= %.4f \n', excentricidad.Eccentricity);</code>
<code>regionprops(IB1,'centroid')</code>	<i>Cálculo de centroide</i>	<code>centroides=regionprops(IB1,'centroid') fprintf('X %.2f; Y %.2f\n', centroides(:,1), centroides(:,2));</code>
<code>regionprops('propiedad1', 'propiedad2')</code>	<i>Cálculo de varias propiedades a la vez.</i>	<code>propiedades= regionprops(L,'Area','Perimeter', 'Eccentricity','Centroid', 'BoundingBox');</code>
<code>bwlabel(IB, vecinos). 4 y 8, por default 8.</code>	<i>Etiquedo de áreas</i>	<code>IOrig=imread('imagen.png'); IGray=rgb2gray(IOrig); %imagen en gris umbral=graythresh(IGray); %umbral IB1=im2bw(IGray,umbral); %binaria [L Ne]=bwlabel(IB); %etiquetado</code>
<code>subplot(2,2,1)</code>	<i>Mostrar matriz de imágenes</i>	<code>subplot(2,2,1);imshow(Iorig); subplot(2,2,2);imshow(IB12); subplot(2,2,3);imshow(Iorig); subplot(2,2,4);imshow(IB2);</code>
		<code>I = imread('rice.png'); background = imopen(I,strel('disk',15)); Ip = imsubtract(I,background); imshow(Ip,[])</code>
	<i>Dibuja Rectángulos</i>	<code>rectangle('Position',propiedades(n).BoundingBox,'EdgeColor','g','LineWidth',2)</code>
	<i>Dibuja Texto en imágenes</i>	<code>text(propiedades(n).Centroid(:,1), propiedades(n).Centroid(:,2),int2str(n), 'Color','g');</code>
<code>int2str</code>	<i>Conversión de entero a cadena</i>	
<code>bwconncomp(B)</code>	<i>Calcula los componentes conectados en una imagen. Trabaja con arrays lógicos o</i>	

Sentencia	Ejemplo
	numéricos
<code>L=labelmatrix(CC)</code>	Etiquetar objetos en una imagen binaria.
<code>bwboundaries(BW)</code>	Calcula los bordes en imágenes binarias
<code>I2 = imcrop(I,rect)</code>	Copiar parte de una imagen utilizando los puntos de coordenadas definidos. %se asume I como una imagen abierta <code>rect=[xmin ymin width height]</code> <code>I2 = imcrop(I,rect)</code>
Separar los canales RGB en distintas matrices	<code>rgbImagen=imread(imagenNombre);</code> <code>canalRojo = rgbImagen(:, :, 1);</code> <code>canalVerde = rgbImagen(:, :, 2);</code> <code>canalAzul = rgbImagen(:, :, 3);</code>
Cambiar valores de pixeles en coordenada específica de una imagen, ejemplo 10, 10 a 255	<code>rgbImagen=imread(imagenNombre);</code> <code>rgbImagen(10,10,:)=255;</code>

8 Historial de cambios

Fecha	Versión	Descripción	Autor
03/11/2016	0,1	Creación y redacción del documento	http://otroblogdetecnologias.blogspot.com
08/11/2016	0,2	Modificación	
24/11/2016	0,3	Modificación	