****

**UNIVERSIDAD TECNOLOGICA EQUINOCCIAL ECUADOR**

**NOMBRE**

Álvarez Ortiz Galo.

**FACULTAD**

Ciencias de la Ingeniería.

**CARRERA**

Ingeniería Mecatrónica.

**MODALIDAD**

Presencial.

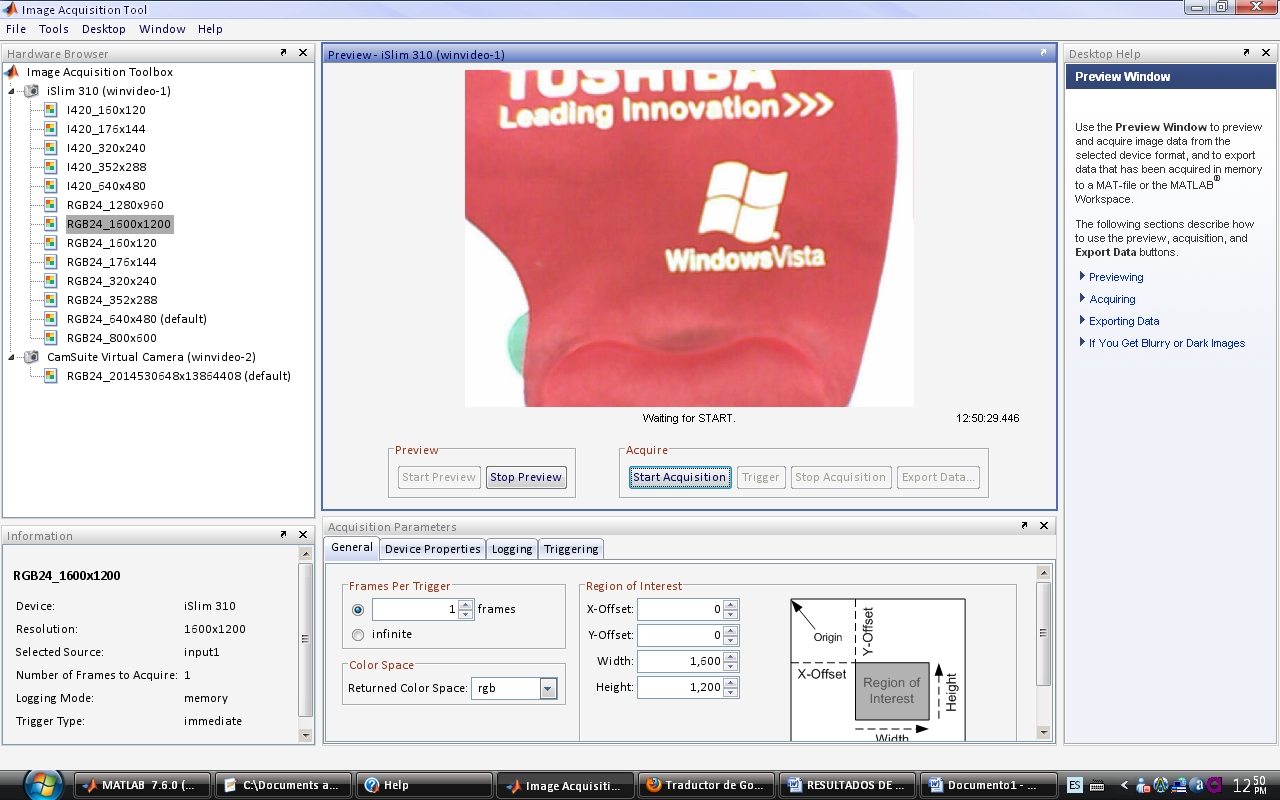
**TEMA**

GUI DE IMAGENES

**COMENTARIO:**

Para comenzar a desarrollar la guía de imágenes empezamos configurando el pixelaje de la webcam.

Se realiza esto, es por el motivo de que no exista problemas al momento de emplear el video y capturar las imágenes, mientras mayor sea el pixelaje de resolución de la webcam, más conveniente es el uso de la misma.



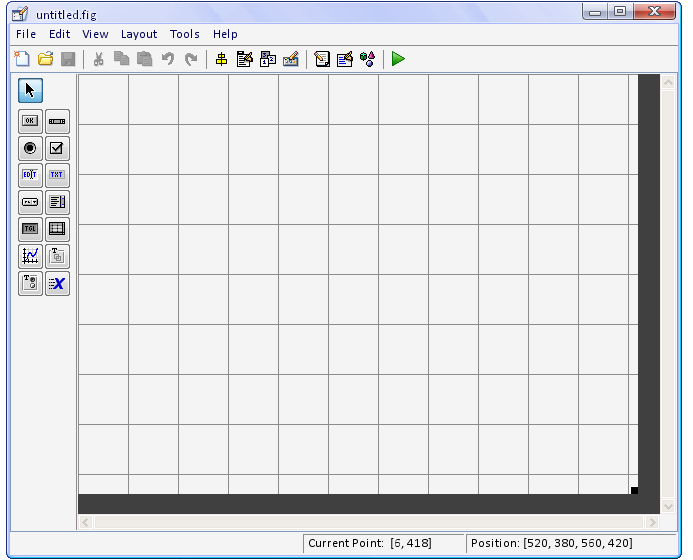
¿Por qué es importante configurar todo esto antes de comenzar?

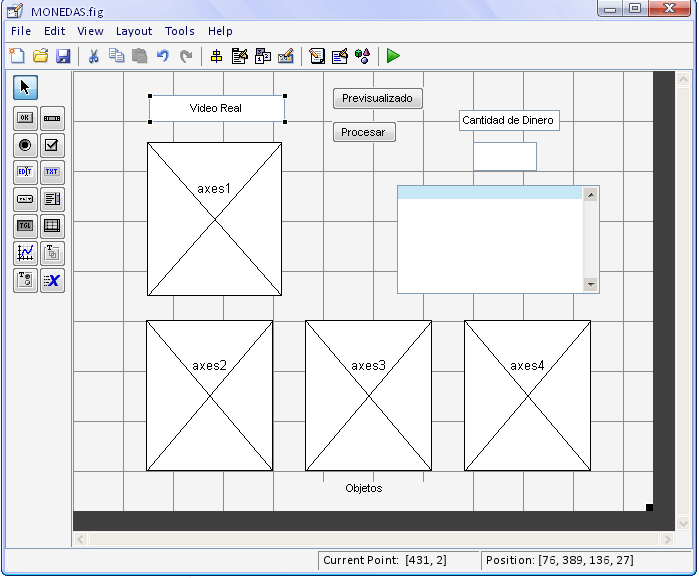
**COMENTARIO:**

Según los inconvenientes que he tenido en las prácticas, es porque si no se tiene activa la webcam, el matlab no reconoce los comandos que se va ejecutando en el editor y no se puede ir yendo paso a paso y seguir avanzando.

Luego de tener bien configurada la webcam, es importante dar clic en stop preview ya que si no se lo hace, al momento de ejecutar el video en la aplicación no se ejecutará y saldrá errores en la línea de comandos

La siguiente parte es configurar el GUI. En esta parte se puede observar y manejar los botones, axes y textos según la interface que se quiera diseñar. También cambiar los nombres a los tag tanto de los axes como de los botones y también la salida de los strings. Se puede poner textos son botones



La configuración del GUI para las monedas la realicé de la siguiente manera

**COMENTARIO**

En la sub barra de tareas que está en la parte izquierda, seleccionamos el botón que dice axes e insertamos en la parte cuadriculada cuatro axes en donde el axes uno cumplirá varias funciones de configuración donde podemos verificar el enfoque, adquirir la imagen, detectar los objetos que en este caso serían las monedas, determinar el nivel de umbralización.

Mientras que en el axes dos se podrá ver la imagen adquirida en escala de grises.

En el axes tres se podrá observar el relleno de la figura

Siguiendo el procedimiento, no se cambiaron ningún nombre a los tags, ya que se han mantenido los mismos

En la parte derecha de los recuadros se ha puesto un listbox en donde saldrá el conteo de las monedas con la descripción de que tipo de moneda es y cuantas hay y un text que ha sido llamado edit1 para que en ese recuadro salga el resultado de la suma de el dinero existente.

Se han agregado dos botones en el GUI. El primero es para pre visualizar el video mientras que el segundo botón se lo ha puesto debajo del primero y se lo ha llamado con el nombre procesar. Este último será el botón que realice todo el proceso de conteo de monedas

**PROGRAMA**

% --- Executes on button press in Previo.

function Previo\_Callback(hObject, eventdata, handles)

% hObject handle to Previo (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

handles.objVideo = videoinput('winvideo');

guidata(hObject,handles);

axes(handles.axes1) %Se llama al eje del realVideo

vidRes = get(handles.objVideo,'VideoResolution');

nBands = get(handles.objVideo,'NumberOfBands');

hImage = image( zeros(vidRes(2), vidRes(1), nBands)); %Numero de columnas del video1 numero de filas 1 y numerode bandas pixelaje

preview(handles.objVideo,hImage);

**COMENTARIO.-** Se comienza ejecutando el código para que aparezca la webcam en video para poder realizar la captura de imágenes. Como se puede observar en las gráficas del GUI al tag del botón se ha puesto el nombre Previo y el String es Previsualizado. Para mi configuración, observo el video con excelente claridad

% --- Executes on button press in Process.

function Process\_Callback(hObject, eventdata, handles)

% hObject handle to Process (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

**COMENTARIO.-** Cuando se da clic en el botón procesar se realizará todo el proceso según el código de programación declarado desde la función Process en el editor para llegar al objetivo final de conteo de monedas clasificándolas en el tipo y la sumatoria total del dinero. Todo se realiza en esta función

%% ADQUIRIR IMAGEN

I\_original = getsnapshot(handles.objVideo);

**COMENTARIO.-** Al adquirir la imagen, es muy importante tener en cuenta las condiciones de iluminación que se tiene para poder continuar con el proceso

%% ESCALA DE GRISES

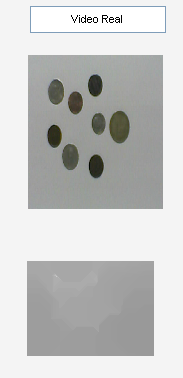
I = rgb2gray(I\_original);

background = imclose(I,strel('disk',100));

axes(handles.axes2)

imshow(background)

**COMENTARIO.-** Esta parte de código es importante porque se configura el background,, se encierra la imagen para tener una tonalidad según el rango de disk que se previsualice mejor. En mi caso el que me ha servido según la condición de iluminación ha sido el valor de 100, se mostrara esta imagen en el axes 2. Observo en mi caso que para la detección de objetos se ha llegado a obtener una escala de grises excelente.



%% DETECCION DE OBJETOS

I2 = imsubtract(background,I);

**COMENTARIO.-** Obtenido el background, el matlab reconocerá y detectará cuantos objetos se tiene. En la imagen puedo observar que matlab detectara los objetos.

%% DETERMINAR EL NIVEL DE UMBRALIZACION

lavel = graythresh(I2);

intensidad = lavel\*255

BW = im2bw(I2, lavel);

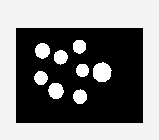
**COMENTARIO.-** El nivel de umbralización no es necesario mostrarlo, por tal motivo, primero se realiza una prueba, en este caso se la ha determinado en el axes 1(en forma abstracta),sin muestra gráfica.

%% RELLENAR LA FIGURA

fill = imfill(BW,'holes');

axes(handles.axes3)

imshow(fill)

**COMENTARIO.-** En esta parte de código matlab realizara el relleno de la imagen según los objetos y la umbralización obtenida. En mi imagen adquirida se puede observar claramente que realiza el relleno de todos los objetos capturados de la imagen.

%% EXTRACCION DE DATOS

[niveles, numObjects] = bwlabel(fill,4)

stats = regionprops(niveles,'Eccentricity','Area','BoundingBox')

areas = [stats.Area]

excentricidad = [stats.Eccentricity];

**COMENTARIO.-** Se realiza esto para que matlab realice cálculos de eccentricidad, área y encierres para los objetos

%% PROCESO ESPECIFICO

**COMENTARIO.-** Para la configuración de los límites de áreas de los objetos lo realice de la siguiente manera: tomé en cuenta dos aspectos importantes, el primero es que la moneda de diez centavos con la de un centavo son parecidas en tamaño por lo que a cada las examiné y coloque en diferentes posiciones para ir tomando valores de áreas de cada una. Realice en cada moneda nueve tomas de medida en diferentes posiciones para sacar el mínimo y el máximo valor. Después me di cuenta que paso el mismo problema con las monedas de un dólar y 25 centavos. Realice el mismo proceso de toma de medidas en diferentes posiciones y obtuve rangos de medida en los cuales no hayan ningún tipo de cruce e intersección. Después de esto prove el resto de monedas una por una de igual manera en algunas tenía problemas de intersección de áreas. Pero al igual que las anteriores hice las tomas de medida una por una..

idxOf25 = find(areas > 6326 & areas < 7211)

stats25 = stats(idxOf25)

axes(handles.axes4)

imshow(I\_original)

hold on;

cantidad = length(stats25)\*0.25;

for idx = 1:length(stats25)

h = rectangle('Position',stats25(idx).BoundingBox,'Curvature',[0.8,0.4],'LineWidth',2,'LineStyle','--');

set(h,'EdgeColor',[0.75 0 0]);

hold on

end

idxOf100 = find(areas > 7508 & areas < 8536)

stats100 = stats(idxOf100)

cantidad = cantidad + length(stats100)\*1

%imshow(I\_original)

for idx = 1:length(stats100)

h = rectangle('Position',stats100(idx).BoundingBox,'Curvature',[0.8,0.4],'LineWidth',2,'LineStyle','--');

set(h,'EdgeColor',[0.75 0 0.75]);

hold on

end

idxOf50 = find(areas > 9725 & areas < 11488)

stats50 = stats(idxOf50)

cantidad = cantidad + length(stats50)\*0.5

for idx = 1:length(stats50)

h = rectangle('Position',stats50(idx).BoundingBox,'Curvature',[0.8,0.4],'LineWidth',2,'LineStyle','--');

set(h,'EdgeColor',[0.75 0.5 0]);

hold on

end

idxOf10 = find(areas > 3200 & areas < 3830)

stats10 = stats(idxOf10)

cantidad = cantidad + length(stats10)\*0.1

for idx = 1:length(stats10)

h = rectangle('Position',stats10(idx).BoundingBox,'Curvature',[0.8,0.4],'LineWidth',2,'LineStyle','--');

set(h,'EdgeColor',[0.75 0 0]);

hold on

end

idxOf5 = find(areas > 4580 & areas < 5613)

stats5 = stats(idxOf5)

hold on;

cantidad = cantidad + length(stats5)\*0.05

for idx = 1:length(stats5)

h = rectangle('Position',stats5(idx).BoundingBox,'Curvature',[0.8,0.4],'LineWidth',2,'LineStyle','--');

set(h,'EdgeColor',[0.75 0 0]);

hold on

end

idxOf1 = find(areas > 3836 & areas < 4524)

stats1 = stats(idxOf1)

hold on;

cantidad = cantidad + length(stats1)\*0.01

for idx = 1:length(stats1)

h = rectangle('Position',stats1(idx).BoundingBox,'Curvature',[0.8,0.4],'LineWidth',2,'LineStyle','--');

set(h,'EdgeColor',[0.75 0 0]);

hold on

end

hold off

set(handles.edit1,'String',num2str(cantidad))

aux = strvcat(strcat('monedas de 5',' =',num2str(length(stats5))),...

strcat('monedas de 10 centavos',' =',num2str(length(stats10))),...

strcat('monedas de 25 centavos',' =',num2str(length(stats25))),...

strcat('monedas de 50 centavos',' =',num2str(length(stats50))),...

strcat('monedas de 1 dolar','=',num2str(length(stats100))),...

strcat('monedas de 1 centavo','=',num2str(length(stats1))))

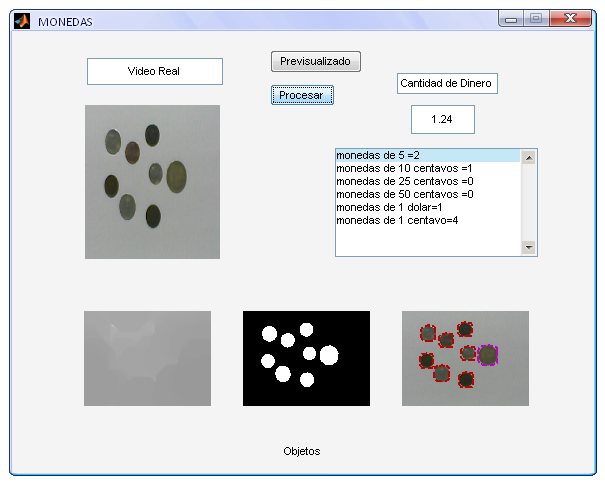
set(handles.listbox1,'String',aux)

**COMENTARIO.-** En esta última parte del código me di cuenta que la cantidad no salía la que en verdad tenía que ser por lo que a cantidad en la anterior parte no estaba aumentando nuevamente cantidad para que vaya sumando los resultados.

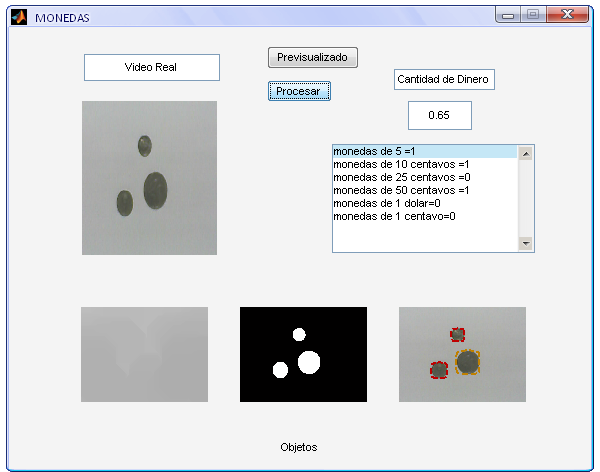
En el edit1 del resultado ya que sí se lo puso de nombre al tag sale la sumatoria total de dinero que se tiene. En mi caso tenía monedas: 4 de un centavo, una de diez centavos, una de un dólar y dos de cinco centavos que en la sumatoria total dio el resultado un dólar con 24 centavos y en el list box aparece la descripción de cuantas monedas hay según el tipo que sean.

Cada una está reconocida, ya que están encerradas según la matriz de código de color que se haya preestablecido en el proceso específico.

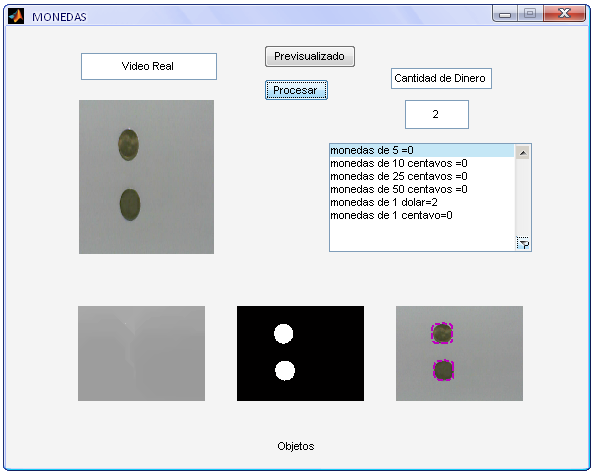
En esta Imagen se puede Observar:



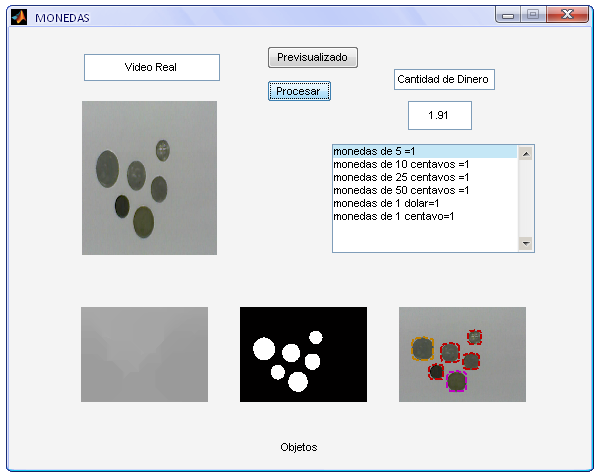
**Otra prueba de funcionamiento**

Una moneda de cincuenta centavos, una de un centavo y una de cinco centavos

Por ejemplo 2 dólares



Todas las Monedas



**COMENTARIO.-** La parte un poco compleja es cuando existe intersección de áreas de tal manera que se descalibra el reconocimiento de cada una de las monedas. Se realizo medidas de áreas en diferentes partes de la zona de espacio.

Excelente procedimiento aprendido, que interesantes clases, me parecen aplicaciones excelentes para el ámbito profesional

Gracias.