

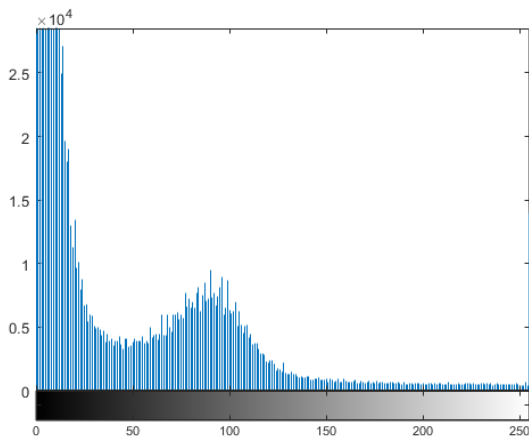
Asignatura: Procesamiento de imágenes**Profesor:** D. Sc. Gerardo García Gil**Alumno:** Jonathan Guillermo Díaz Magallanes.**Registro:** 19310153. **Ciclo:** 2022-B**Ingeniería en Desarrollo de Software****Centro de Enseñanza Técnica Industrial (CETI)**

Introducción

A lo largo de éste documento se documentará una práctica la cual consiste en usar las funciones ya realizadas del histograma simple, histograma acumulado y histograma lineal de una imagen ecualizada, pero ésta vez las vamos a utilizar por medio de Simulink, proporcionandonos la habilidad de hacer el cálculo en vivo con la cámara de video en lugar de una imagen.

Histograma

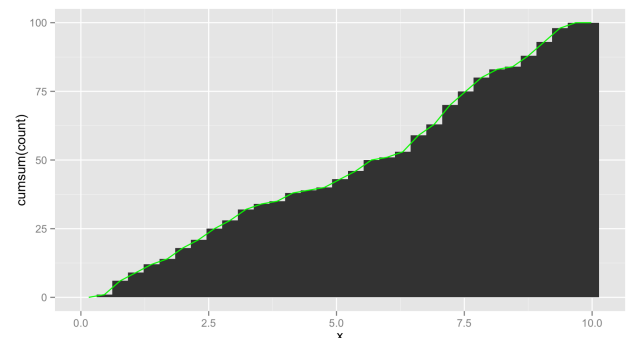
El Histograma muestra importantes características de una imagen, como por ejemplo el contraste y el rango dinámico, los cuales son problemas atribuidos a la captación de la imagen que es necesario resolver para que puedan ser analizadas con mejor claridad propiedades de la imagen en bloques de procesamiento posterior.



Algunas de las características que se pueden tomar de una imagen son el contraste y la dinámica, problemas que son producidos durante la toma de la imagen y generan consecuencias para las siguientes etapas de procesamiento.

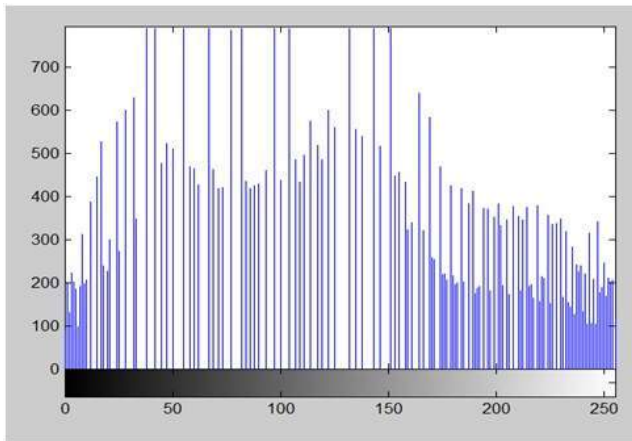
Histograma acumulativo

El histograma acumulativo es una variante del histograma normal, el cual refleja información importante para la realización de operaciones de pixel por pixel en imágenes para realizar ciertas operaciones, un ejemplo de ellas es la acción de equilibrar un histograma.



Histograma lineal

La ecualización de histograma se utiliza para mejorar el contraste. No es necesario que el contraste siempre se incremente en esto. Puede haber algunos casos en los que la ecualización del histograma sea peor. En esos casos se reduce el contraste.



Desarrollo

Lo primero que se realizó fué el diagrama de bloques en Simulink.

El primer bloque que se necesitó es el que se encarga de obtener la imagen de video capturada por nuestra cámara web, el cual lo obtenemos por el paquete llamado *image acquisition toolbox*, dicho paquete se llama *from video device*

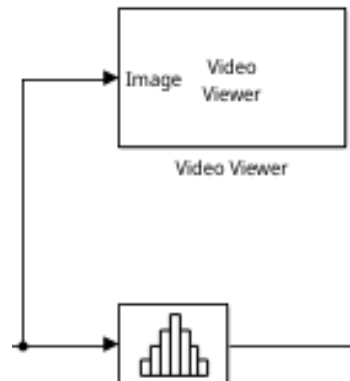


De éste bloque pasaremos la salida a otro donde convertiremos la escala del vídeo a escala de grises

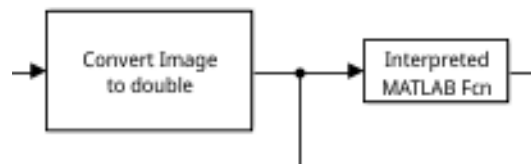


Seguido conectaremos dos bloques a la misma salida, uno de ellos para mostrar la salida del video (Video

Viewer) y un generador de histograma (2D histogram)



La salida del histograma la conectaremos a un convertidor a datos de tipo double, donde tomara los valores del histograma a un tipo de datos que el interpretador los pueda aceptar

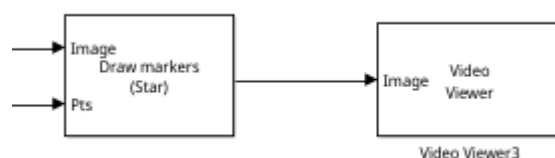


En el interpretador es donde mandaremos a llamar nuestras funciones de Matlab donde estaremos usando dos funciones:

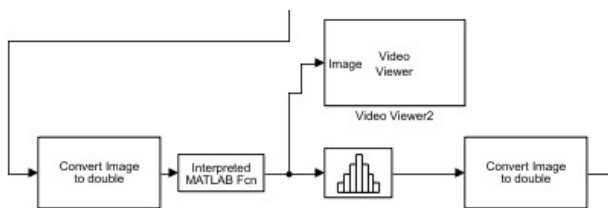
Colecta: la cual es el código que recopila datos para un histograma simple.

Concatena: la cual es el código para el histograma acumulado.

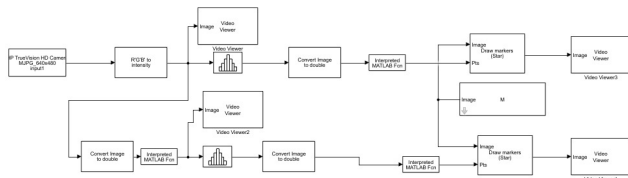
Finalmente tendremos dos bloques que juntos nos van a mostrar el histograma en tiempo real. Éstos bloques son el Draw maker el cual va a dibujar el histograma como un dibujo en un pintarrón y con el video viewer vamos a mostrarlo en una ventana.



Para agregar el apartado de la ecualización de la imagen tendremos que añadir una sección completa conformada por un image to double el cual tomaremos como entrada la imagen del video en escala de grises, para seguidamente pasar la salida a un interpretador donde añadiremos la función del ecualizado de la imagen “vision2” y al final de la misma salida un video viewer para poder visualizar el cambio del contraste con referencia a la versión anterior y un 2-D histogram donde la salida del histograma irá a un image to double para dirigirlo al acumulado.



Al final con todos los bloques usados obtendremos el siguiente diagrama:



Cuando ejecutamos el programa nos lanzará cuatro ventanas; la imagen de la videocámara, la misma imagen con el contraste mejorado, la del histograma acumulado y la del histograma simple:

Imagen en escala de grises

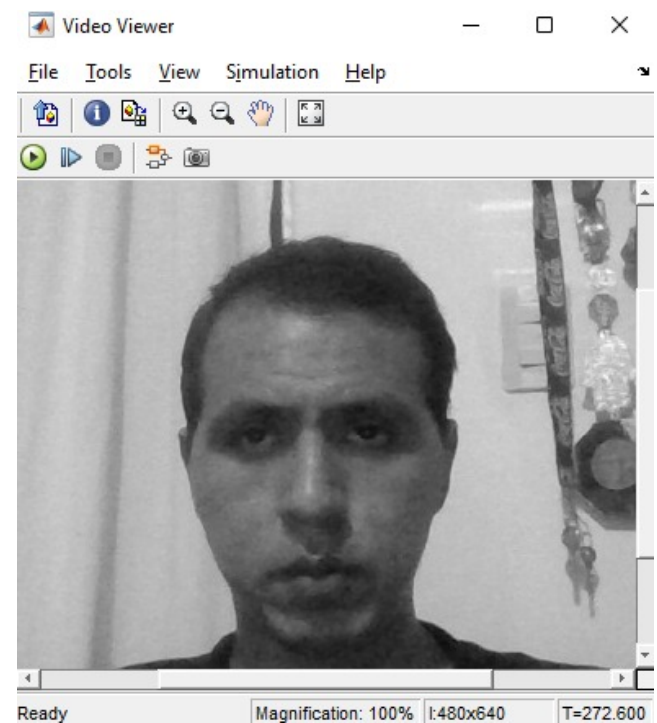
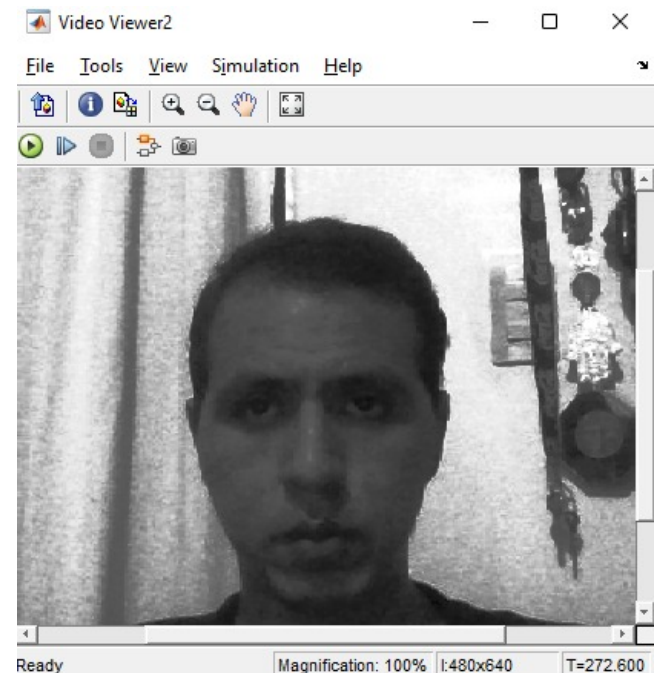
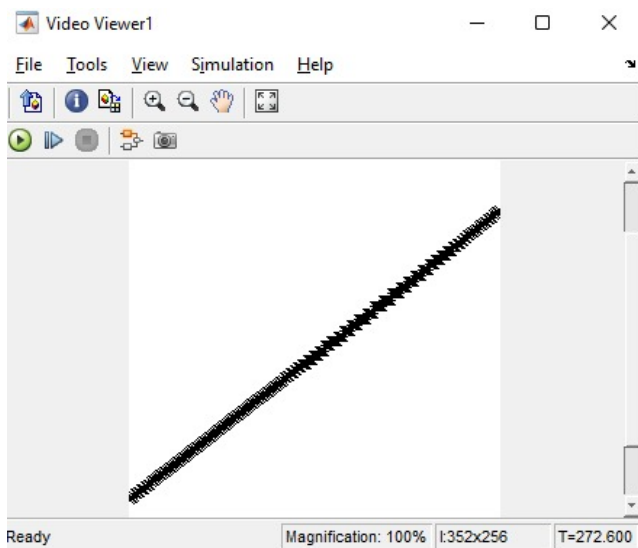


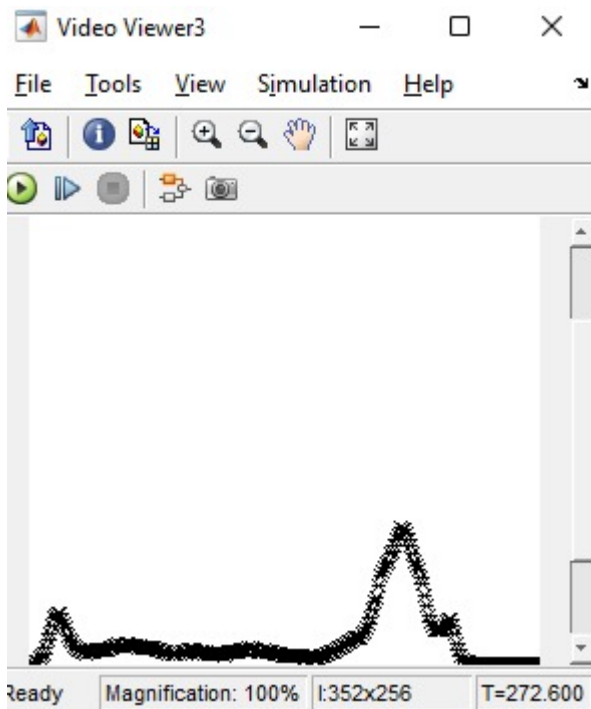
Imagen ecualizada



Histograma lineal



Histograma simple



Código

```
function union = colecta(u)
```

```
num = 1:256;
```

```
val=288-(u*3000);
```

```
union(:,1)=num;
```

```
union(:,2)=val;
```

```
end
```

```
function union = concatena(u)
```

```
num=1:256;
```

```
va=0;
```

```
for v=1:256
```

```
h(v)=va+u(v);
```

```
va=h(v);
```

```
end
```

```
val=288-(h*200);
```

```
union(:,1)=num;
```

```
union(:,2)=val;
```

```
end
```

```
function IM = vision2(u)
```

```
sum=0;
```

```
[datos,x]=imhist(u);
```

```
[R,C] = size(u);
```

```
IM=zeros(R,C);
```

```
H=zeros(1,256);
```

```
for cont=1:256
```

```
H(cont)=datos(cont)+sum;
```

```
sum=H(cont);
```

```
end
```

```
for r=1:R
```

```
for c=1:C
```

```
temp=floor(u(r,c)*255);
```

```
IM(r,c)=H(temp+1)/(R*C);
```

end

end

end

Conclusión

En el desarrollo de ésta práctica no se tuvieron problemas por parte de la creación del diagrama de Simulink o al momento de escribir el código, sino con Matlab en sí. Debido a que uso el sistema operativo GNU/Linux, Simulink no funciona del todo bien debido a la incompatibilidad o falta de librerías que se pueda tener en el sistema. Después de pasar días tratando de poner en funcionamiento Simulink se decidió mejor hacer la instalación en Windows debido a que Simulink no detectaba la cámara.

Referencias

- *Cumulative Histogram - MIPAV*. (n.d.).
MIPAV. Retrieved August 25, 2022, from
https://mipav.cit.nih.gov/pubwiki/index.php/Cumulative_Histogram
- *Ecualización de histograma*. (n.d.). EDU.
Retrieved August 25, 2022, from
<https://tutoriales.edu.lat/pub/dip/histograma-equalization/ecualizacion-de-histograma>