

Introducción al Procesamiento Digital de Imágenes

Practica 1

Cristian Galli L.U. 538/11

cristiangalli88@gmail.com

Esta hoja intencionalmente en blanco.

Esta aplicación usa OpenCV.

Instalación OpenCV Ubuntu.

Guia de http://docs.opencv.org/2.4/doc/tutorials/introduction/linux_install/linux_install.html

```
sudo apt-get install build-essential
sudo apt-get install cmake git libgtk2.0-dev pkg-config libavcodec-dev libavformat-dev
libswscale-dev
git clone https://github.com/opencv/opencv.git
cd opencv
mkdir release
cd release
cmake -D CMAKE_BUILD_TYPE=RELEASE -D CMAKE_INSTALL_PREFIX=/usr/local ..
make
sudo make install
```

Note: Use `cmake -DCMAKE_BUILD_TYPE=RELEASE`
`-DCMAKE_INSTALL_PREFIX=/usr/local ..` , without spaces after -D if '`cmake -D`
`CMAKE_BUILD_TYPE=RELEASE -D CMAKE_INSTALL_PREFIX=/usr/local ..`' do not work.

Para Compilar

En la carpeta Source abrir una terminal y hacer:

```
g++ -std=c++11 *.cpp -o <nombre salida> `pkg-config --cflags --libs opencv`
```

ejemplo

```
g++ -std=c++11 *.cpp -o main `pkg-config --cflags --libs opencv`
```

Si al correr el programa tira un error parecido al siguiente

```
./main: error while loading shared libraries: libopencv_imgcodecs.so.3.3: cannot open
shared object file: No such file or directory'
```

hacer en la terminal

```
export LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:/usr/local/lib
sudo ldconfig
```

Y probar compilando otra vez.

Para correrlo

En la terminal hacer

```
./<nombre salida>
```

En el ejemplo
./main

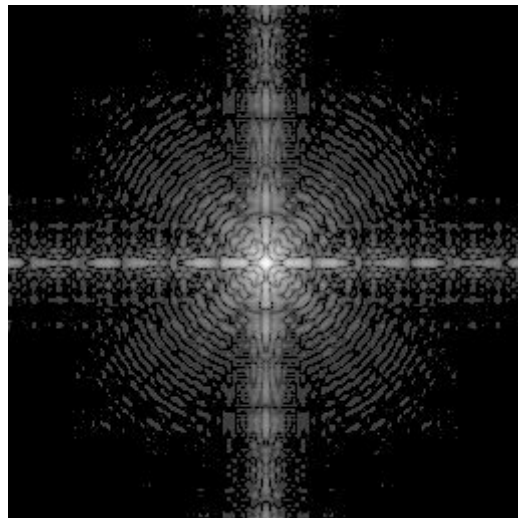
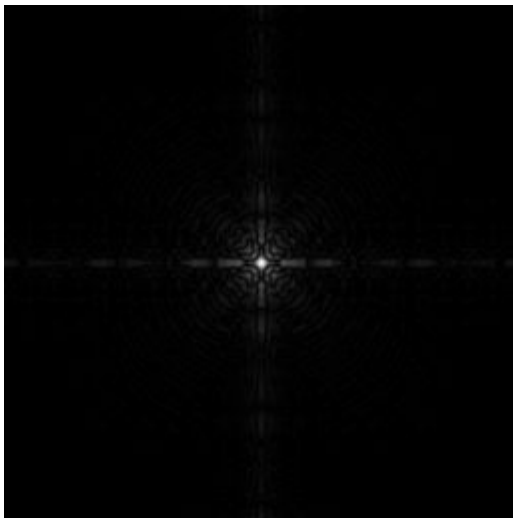
Luego ingresar el nombre de imagen de entrada incluyendo extensión.
Luego ingresar el nombre de imagen de salida incluyendo extensión.

Ejercicios

Compresion del rango dinamico

Para 'Compresion del rango dinamico' usar el método '*compresionDelRangoDinamico(int nivelDeGrisDeimagenDeSalida)*' de '*Imagen*'.

Ejemplo:



Negativo

Usar el método *negativo()* de *Imagen*.

Ejemplo:



Umbral

Método *umbral(int u)*.

Ejemplo:

umbral(100)



Planos de Bits

Método *planosDeBits()*.

Ejemplo

Del 0 al 7

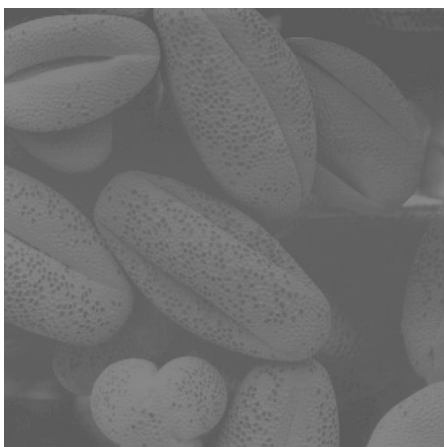


Histograma

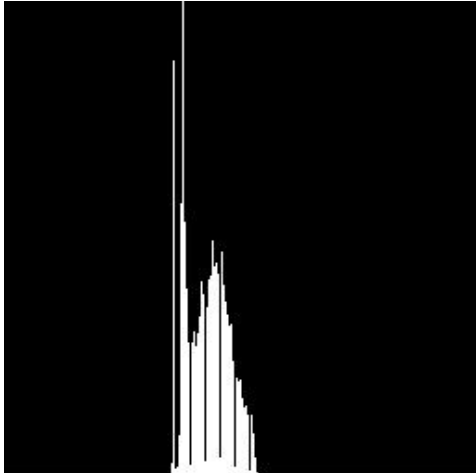
Método *histograma()*.

Ejemplo

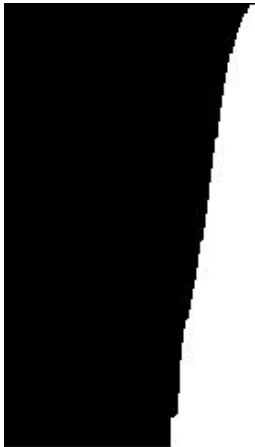
Imagen



Histograma



Acumulado



Contraste

contraste().

La idea es usar la siguiente función partida para que los valores que están en el intervalo más chico vayan a valores más chicos, los valores del medio se queden igual y los del intervalo más grande aumenten su valor.

Min = mínima intensidad;

Max = máxima intensidad;

Rango = rango de valores entre Min y Max;

para cada posición de la imagen

Si el valor de esta posición está dentro de $\frac{1}{5}$ del Rango

asigno $\frac{1}{5}$ del valor

Si está entre $\frac{1}{5}$ y $\frac{2}{5}$ asigno $\frac{2}{5}$ del valor

Si está entre $\frac{2}{5}$ y $\frac{3}{5}$ lo dejo igual

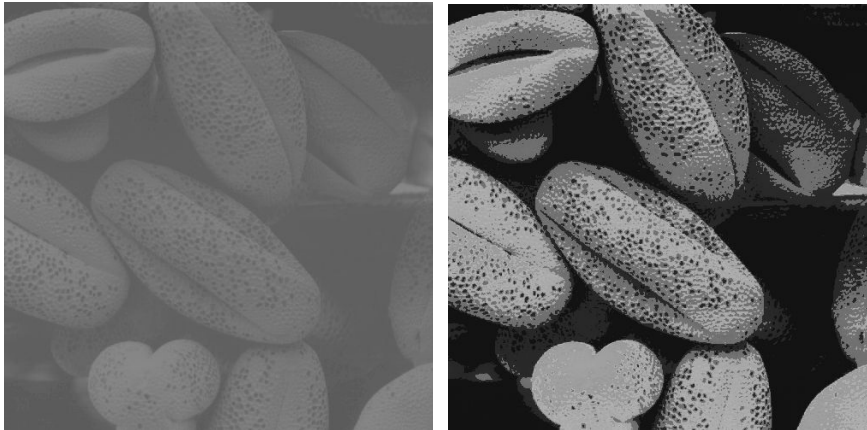
Si esta dentre de $\frac{3}{5}$ y $\frac{4}{5}$ asigno $\frac{7}{5}$ del valor saturando en 255

Si esta entre $\frac{4}{5}$ y 1 asigno $\frac{6}{5}$ del valor saturando en 255

fin para

Ejemplo

Imagen original vs imagen con contraste



Histograma original vs histograma de imagen con contraste



Acumulado de histograma original vs acumulado de histograma de imagen con contraste



Ecualización

Método *ecualizarConHistograma(const Histograma &histograma)* pasando como parámetro el histograma con el cual ecualizar.

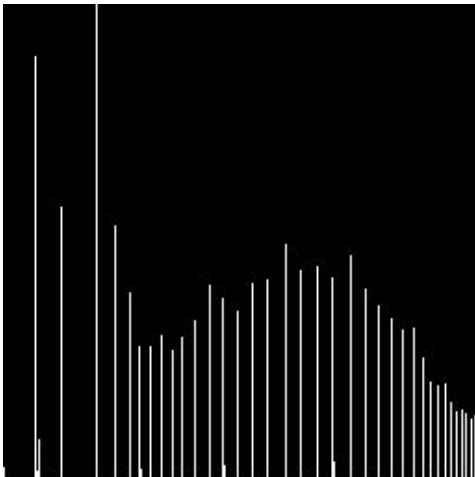
Ejemplo

img1.ecualizarConHistograma(h.acumuladoEnIntervaloCeroYCantidadDeElementosMenosUno()) con *img1* la imagen y *h* el histograma de la imagen.

Imagen ecualizada



Histograma de imagen ecualizada



Acumulado de histograma de imagen ecualizada



Doble Ecualización

Aplicar ecualización a la imagen ya ecualizada utilizando su nuevo histograma:

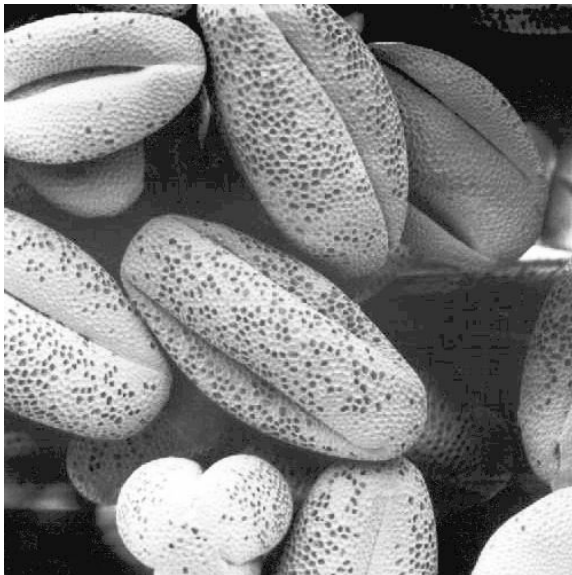
```
ecualizada.ecualizarConHistograma(histoEcu.acumuladoEnIntervaloCeroYCantidadDeElementosMenosUno());
```

con *ecualizada* igual a la imagen ya ecualizada (obtenida en el punto anterior) e *histoEcu* igual al histograma de la imagen ya ecualizada (obtenida en el punto anterior).

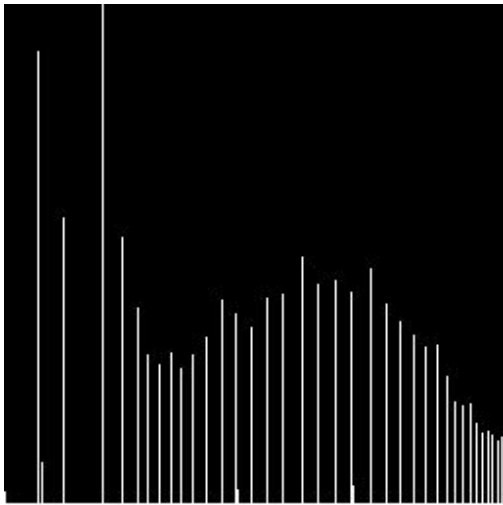
Los resultados no cambian ya que el histograma ya tenía una distribución uniforme.

Ejemplo

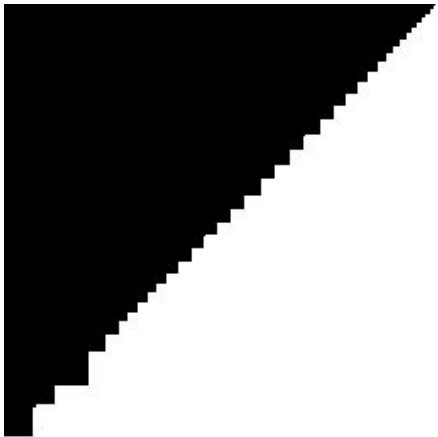
Imagen doble ecualizada



Histograma de imagen doble ecualizada



Acumulado de histograma de imagen doble ecualizada



Especificación del histograma para obtener una distribución normal con media $L/2$ y desvío $L/4$

```
Histograma normalHisto = Histograma(256, DistribucionNormalFunctor(256.0 / 2.0, 256.0 / 4.0));
Histograma normalHistoAcum = normalHisto.acumulado();
Histograma especificadoConNormal =
h.especificarParaObtenerDistribucionAcumulada(normalHistoAcum);
Imagen normalEculmg = img1.ecualizarConHistograma(especificadoConNormal);
```

Primero se obtiene un histograma que representa una distribución normal.

Luego obtener el acumulado.

Luego especificar el histograma original con el acumulado del de distribución normal.

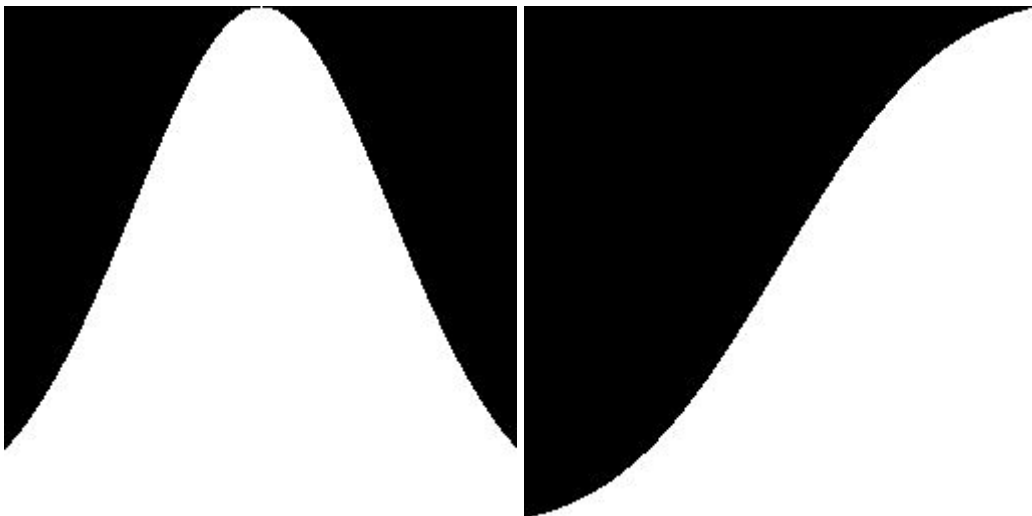
Y por último ecualizar la imagen con este último histograma obtenido.

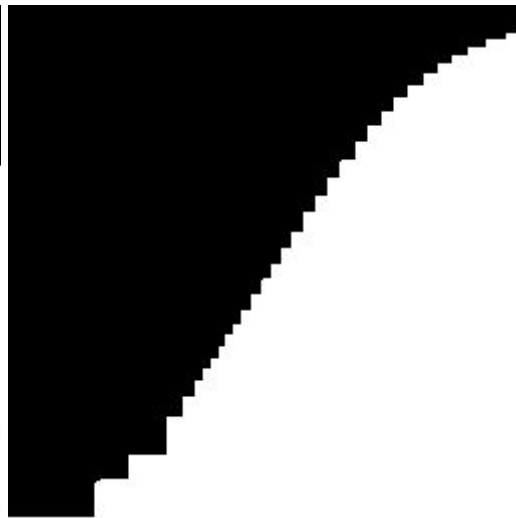
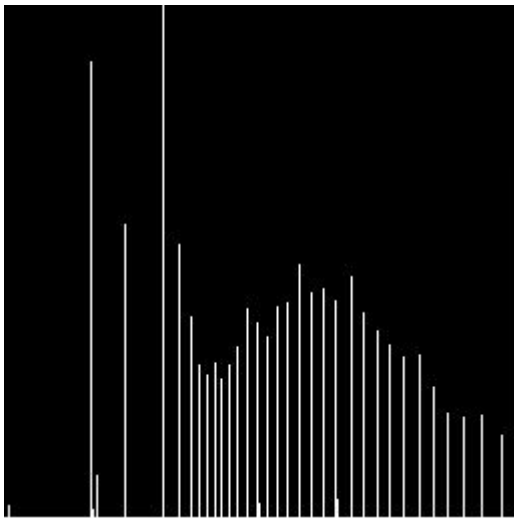
Ejemplo

Imagen ecualizada



Arriba histograma de la normal junto a su acumulada.
Medio histograma especificado con normal junto a su acumulada.
Abajo histograma de la imagen ecualizada junto a su acumulada.





Modificación de histograma

Histograma modificado $Hist1 = h.histogramaModificadoComoUniforme(1);$

Histograma modificado $Hist2 = h.histogramaModificadoComoUniforme(2);$

con h el histograma original.

Ejemplo

Imagen original



Histograma modificado junto a su acumulada ($\lambda = 1$)

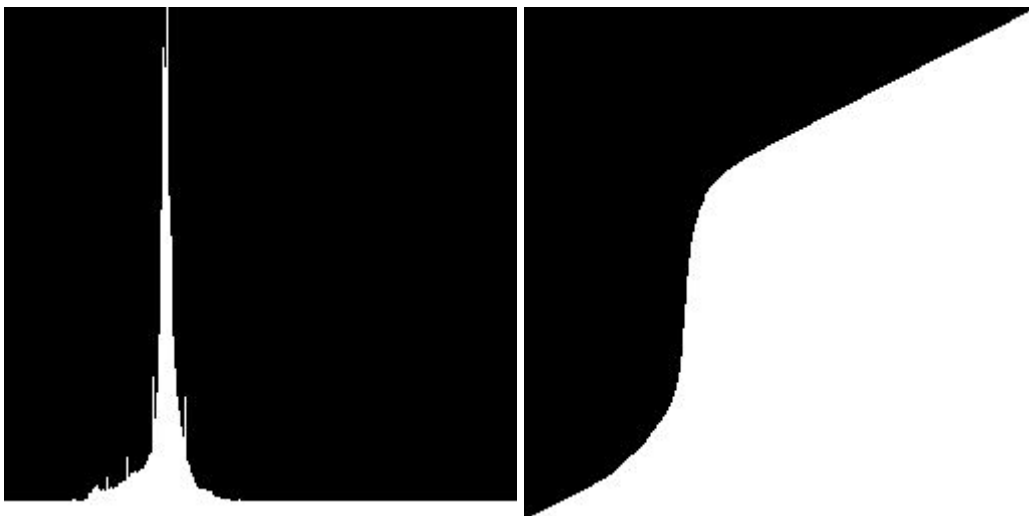
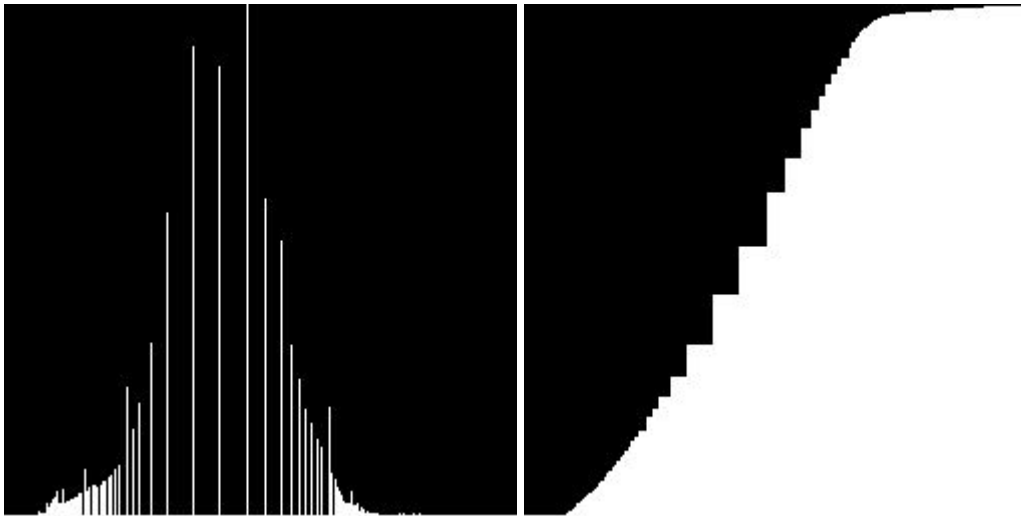


Imagen ecualizada con el histograma modificado ($\lambda = 1$)



Histograma de la imagen ecualizada con el histograma modificado ($\lambda = 1$)



Histograma modificado junto a su acumulada ($\lambda = 2$)

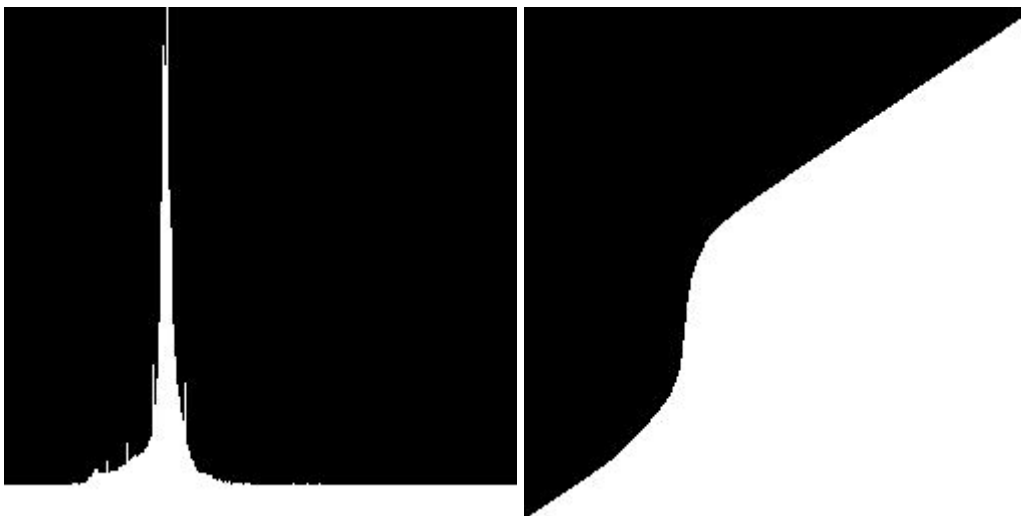


Imagen ecualizada con el histograma modificado ($\lambda = 2$)



Histograma de la imagen ecualizada con el histograma modificado ($\lambda = 2$)

