

# Master: Ciencia de Datos e Ingeniería de Computadores

Curso: Visión por Computador

Rosa M<sup>a</sup> Rodríguez Sánchez

Dpto. Ciencias de la Computación e I. A.

E.T.S. de Ingenierías Informática y de Telecomunicaciones

Universidad de Granada



## Transformaciones del histograma.

### Índice de contenido

1. Introducción.....	2
2. Operaciones aritméticas simples.....	2
2.1. Sumar una imagen y un escalar.....	2
2.2. Producto de una imagen y un escalar.....	2
3. Transformaciones genéricas y función de transferencia.....	3
4. La ecualización del histograma.....	4
5. Ejercicios.....	5

## 1. Introducción

En esta práctica veremos algunas técnicas para manipular el histograma y mejorar el contraste de las imágenes.

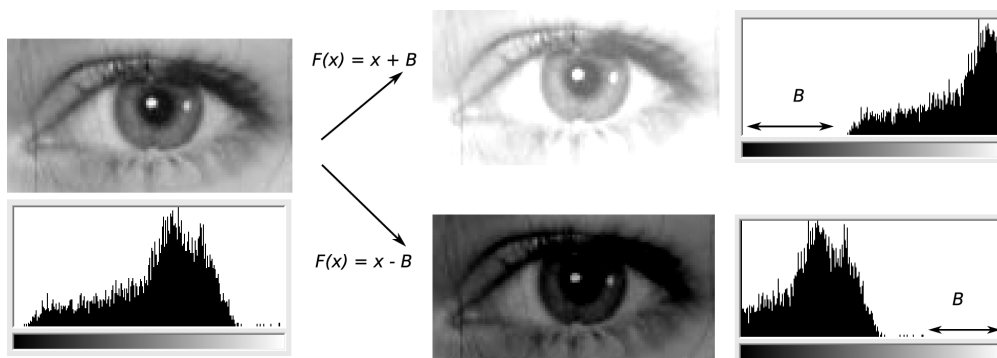
## 2. Operaciones aritméticas simples

### 2.1. Sumar una imagen y un escalar

Esta operación aumenta el brillo de manera uniforme. El efecto sobre el histograma es que lo desplaza hacia la derecha. Posiblemente muchos píxeles serán saturados al máximo valor.

```
>> nueva = imagen + B
```

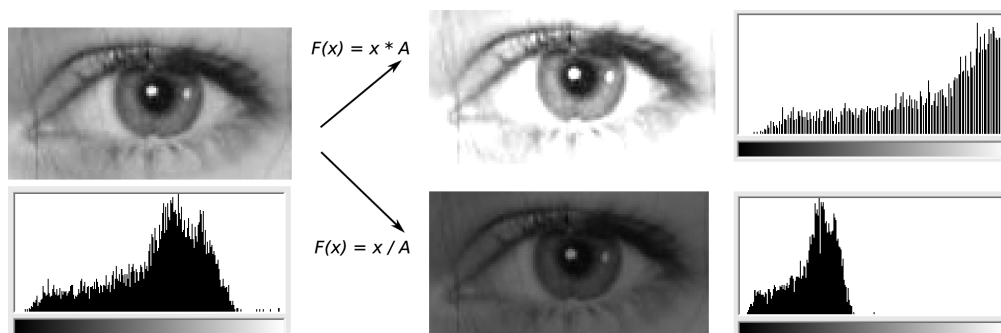
Si B es negativo o bien restamos en lugar de sumar, el efecto es una disminución de brillo y el consiguiente desplazamiento hacia la izquierda del histograma. En este caso la saturación se producirá hacia el cero.



### 2.2. Producto de una imagen y un escalar

En este caso, al multiplicar también obtendremos una imagen con más o menos brillo (dependiendo de si el factor es mayor o menor que uno). La diferencia con la suma es que el aumento es proporcional al valor de gris multiplicado. El efecto sobre el histograma es un estiramiento (si el factor es mayor que uno) o un aplastamiento (si el factor es menor que uno).

```
>> nueva = imagen * A
```



### 3. Transformaciones genéricas y función de transferencia

Cualquier transformación sobre el histograma se puede representar mediante su función de transferencia. Esta función define la relación entre los valores de gris de la imagen original y los valores por los que van a ser sustituidos en la imagen resultante.

Las operaciones aritméticas que hemos visto antes se pueden modelar con funciones de transferencia lineales:

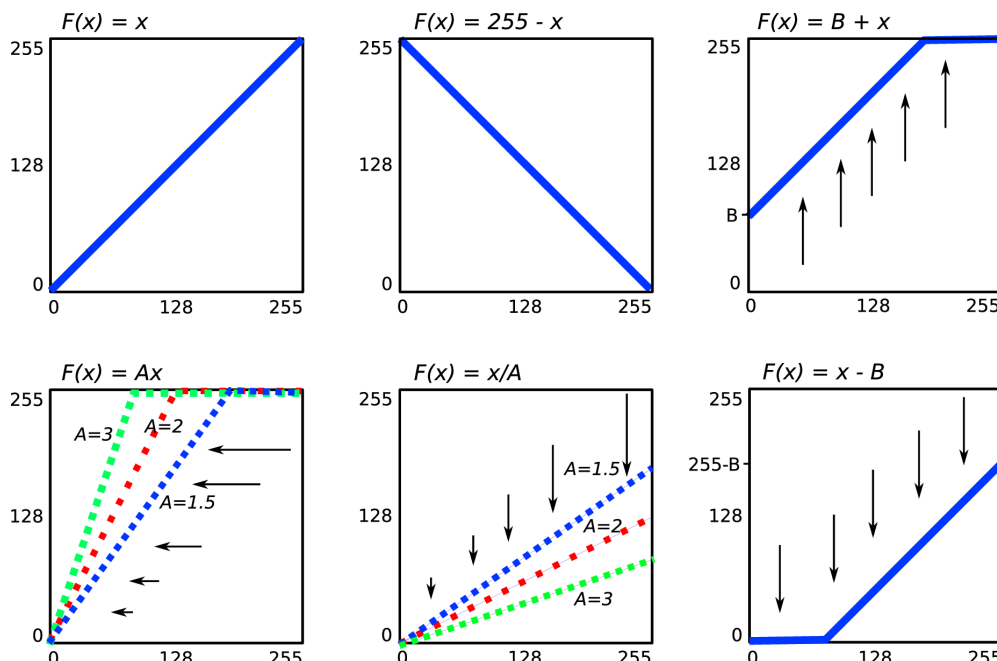
- Suma:  $F(x) = B + x$
- Multiplicación:  $F(x) = A * x$

donde A es una constante. En general, cualquier transformación lineal es de la forma:

$$F(x) = A * x + B$$

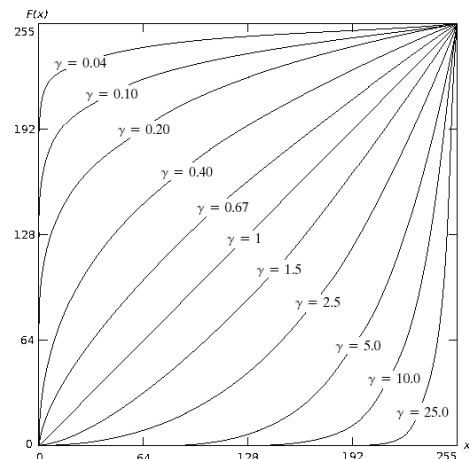
La inversión de los niveles de gris se consigue mediante  $F(x) = MAX - x$ , donde MAX es el valor máximo de gris que se puede representar en la imagen (si la imagen es double será 1.0, si la imagen es de clase uint8 el máximo es 255, etc.). En Matlab disponemos de la función `imcomplement` para calcular la imagen complementaria:

```
>> negativo = imcomplement(imagen)
```



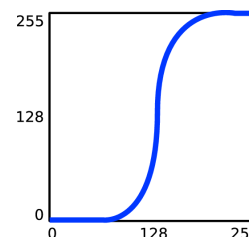
También es frecuente usar funciones no lineales para mejorar algunos resultados y, a veces, para evitar la saturación de los valores. Por ejemplo:

- Transformación de gama:  $F(x) = c * x^\gamma$ . Esta función (u otras similares) se utiliza para aplicar lo que se conoce como corrección gamma en monitores y otros dispositivos de visualización o captura de imágenes. A la derecha vemos el aspecto que presenta esta transformación para diferentes valores de gamma.
- Se pueden aplicar más tipos de funciones como logarítmicas, exponenciales o raíces para obtener efectos similares a los que se obtienen con la transformación de gama.
- Se pueden componer, por trozos, varias funciones. Por ejemplo, para valores de gris inferiores a 128 se puede aplicar una función exponencial y para el resto una logarítmica.



Una función muy utilizada para la mejora del contraste es la conocida como stretching (estiramiento). Esta función tiene la forma que vemos en la figura de la derecha:

- Los valores inferiores a un cierto umbral se anulan.
- Los superiores a otro umbral se saturan al máximo.
- Los valores intermedios se transforman con alguna función.



En Matlab disponemos de la función `imadjust`, que usada conjuntamente con `stretchlim` permite hacer estiramientos del histograma.

## 4. La ecualización del histograma

Para ecualizar el histograma disponemos de la función `histeq`. Podemos usarla de varias formas:

```
>> q = histeq(b);
```

Así, tomamos la imagen `b`, ecualizamos su histograma y devolvemos la imagen ecualizada. Otra forma de usar la función es:

```
>> q = histeq(b,N);
```

el resultado, en este caso, es una imagen con `N` niveles de gris (ecualizados). Otra forma de uso es la siguiente:

```
>> q = histeq(b,V);
```

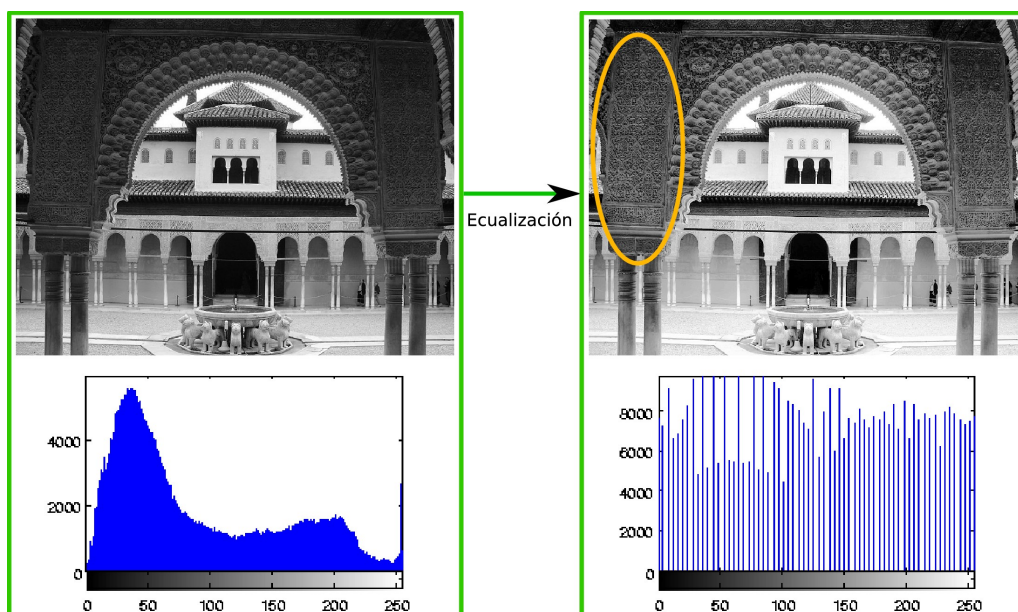
en este caso, `V` es un vector con valores que modeliza el histograma al que deseamos que se parezca el histograma de nuestra imagen (véase la documentación de matlab para más detalles sobre el método que se aplica).

Si deseamos obtener los valores de la transformación del nuevo histograma, podemos almacenar el resultado de la función:

```
>> [N T] = histeq(b);
```

`T` define la transformación que se ha de aplicar para ecualizar la imagen. `N` es la imagen ecualizada.

Normalmente, el resultado de la ecualización produce una mejora del contraste, tal como podemos ver en la siguiente figura. Podemos observar que las zonas oscuras se ven bastante mejor tras la ecualización:

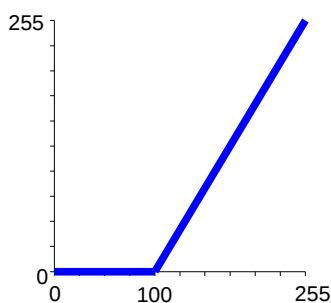
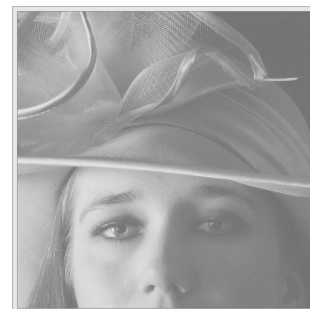


Obviamente, esta técnica no es perfecta. Como podemos ver en el ejemplo anterior, algunas zonas de la imagen no están suficientemente contrastadas (el interior de la puerta o de las ventanas del edificio de enfrente). Por contra, algunas otras zonas aumentan demasiado la luminosidad para poder compensar las zonas oscuras (por ejemplo, los leones de la fuente).

## 5. Ejercicios

1. Cargar la imagen **mujer.jpg** y realiza las siguientes operaciones:

- Mejora el contraste usando únicamente operaciones aritméticas (+, -, \*, /).
- Usa `imadjust` para mejorar el contraste. Usa la función `stretchlim`.
- Usa `imadjust` para aplicar una función de transferencia de tipo gamma. Comprueba el efecto que produce la transformación en la imagen y en el histograma.
- Usa `imadjust` para aplicar la siguiente función de transferencia:



- Finalmente, ecualiza la imagen.

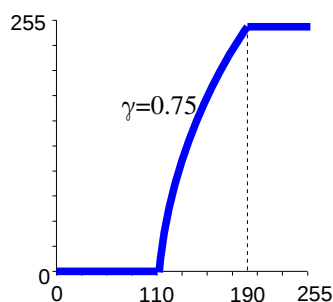
2. Coge una imagen en color y utiliza `imadjust` o `histeq` para mejorar el contraste. Considera estas posibilidades:

- Ecualizar por separado cada banda R, G, B.
- Ecualizar por separado cada banda H, S, V.

¿Cómo podemos conseguir un buen ecualizado para imágenes en color? Usa **danza.ppm** para contestar a esta pregunta.



3. Usa `imadjust` para aplicar la siguiente función de transferencia a la imagen **campo.ppm**:



4. Usa la función `adapthisteq` sobre la imagen **mujer.jpg** y analiza el resultado.

5. Aplica distintas técnicas de mejora de contraste sobre la imagen siguiente y compara los

resultados:

