

Operaciones de pixel

Temario

- Operaciones de pixel básicas:
 1. Modificar la intensidad de la imagen.
 2. Invertir la imagen.
 3. Operador umbral.
 4. Autocontraste.
 5. Autocontraste modificado.
 6. Operador logaritmo.
 7. Operador exponencial.
 8. Corrección gamma (operador potencia).

Operaciones de pixel

- Las operaciones de pixel realizan una modificación de los valores de los pixeles sin cambiar el tamaño, la geometría o la estructura local de la imagen.
- Cada nuevo valor de pixel $b = I'(u, v)$ depende exclusivamente del valor anterior $a = I(u, v)$ en la misma posición.

Modificar la intensidad de la imagen

- Consiste en cambiar los valores de los píxeles.
- Ejemplo: incrementar la intensidad en un 50% se expresa así:
- $f(a) = a * 1.5$
- Donde a es la intensidad original del píxel.
- También se puede incrementar la intensidad en, por ejemplo, 10 unidades:
- $f(a) = a + 10$

Modificar la intensidad en OpenCV

- La función `addWeighted(imagen1, a, imagen2, b, c)` regresa:
- $a * \text{imagen1} + b * \text{imagen2} + c$.
- Se puede usar para modificar la intensidad de la imagen1 poniendo b y la imagen2 a ceros.

Ejemplo

Original Image



Contrast=2.3, Brightness=10



Invertir una imagen

- Para un valor de pixel $a = I(u, v)$ en el rango $[0, a_{max}]$ la operación de pixel es
- $f_{inv}(a) = a_{max} - a$
- Para una imagen en escala de grises de 8 bits $a_{max} = 255$.
- En una imagen a color, la inversión se hace para cada canal.

Ejemplo

255	100	0	90
200	40	25	140
152	77	220	192
45	88	25	110

Imagen original

0	155	255	165
55	215	230	115
103	178	35	63
219	167	230	145

Imagen invertida

Invertir una imagen con OpenCV

- Se utiliza la función `cv.bitwise_not(image)`.

Invertir una imagen en tonos de gris

Original Image



Inverted Image



Invertir una imagen a color

Original Image



Inverted Image



Operador umbral

- A cada pixel el operador umbral le asigna uno de dos valores de intensidad fijos, a_0 o a_1 , dependiendo de un umbral q que suele ser una constante.
- $$f_{\text{threshold}}(a) = \begin{cases} a_0 & \text{para } a < q \\ a_1 & \text{para } a \geq q \end{cases}$$
- Con $0 < q \leq a_{\text{max}}$.
- Una aplicación común es binarizar una imagen con los valores $a_0 = 0$ y $a_1 = 255$.

Operación umbral en OpenCV

- Se utiliza la función `cv.threshold()`.

Ejemplo

Original Image



Binary Image



Contraste

- Es la diferencia entre los valores máximo y mínimo de los píxeles de la imagen ($a = a_{max} - a_{min}$).
- Idealmente, en una imagen en tonos de gris $a_{min} = 0$ y $a_{max} = 255$ y el contraste es a es 255.
- A este tipo de imagen se le llama de contraste completo (*full-contrast*).

Contraste



Bajo contraste



Alto contraste

Autocontraste

- Mapea los pixeles linealmente para que se ocupe todo el rango.
- Se utiliza para mejorar el contraste.
- Suponer que a_{lo} y a_{hi} son los valores mínimos y máximos en una imagen dada.
- Suponer que el rango de intensidad posible es $[a_{min}, a_{max}]$.

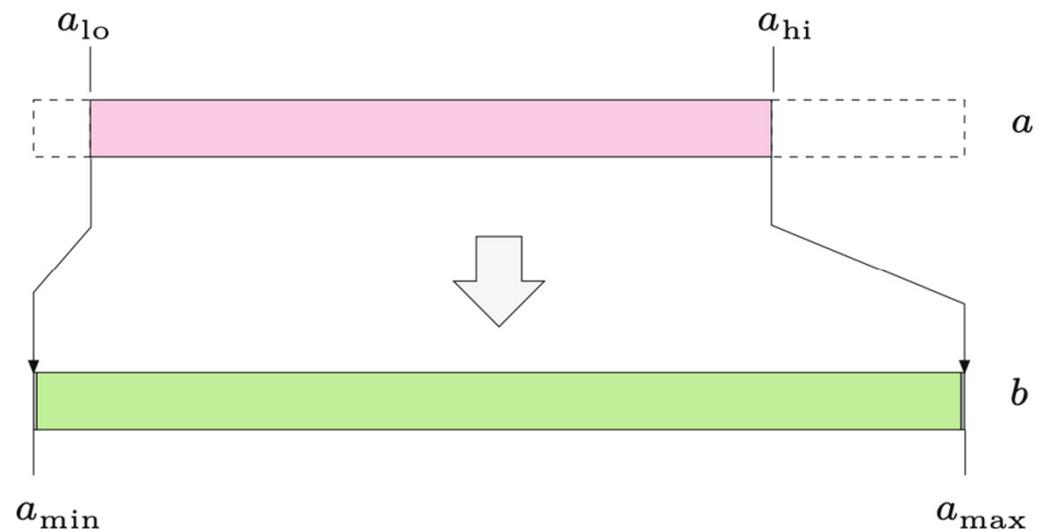
Autocontraste

4 POINT OPERATIONS

Fig. 4.4

Auto-contrast operation according to Eqn. (4.7).

Original pixel values a in the range $[a_{lo}, a_{hi}]$ are mapped linearly to the target range $[a_{min}, a_{max}]$.



Autocontraste

- La intensidad final $f_{ac}(a)$ está dada por:
- $$f_{ac}(a) = a_{min} + (a - a_{lo}) \cdot \frac{a_{max} - a_{min}}{a_{hi} - a_{lo}}$$
- Se supone que $a_{hi} \neq a_{lo}$.
- Para una imagen en escala de grises de 8 bits, $a_{min} = 0$, $a_{max} = 255$ y la ecuación se simplifica:
- $$f_{ac}(a) = (a - a_{lo}) \cdot \frac{255}{a_{hi} - a_{lo}}$$
- En general, este método puede tanto ampliar como reducir el contraste.

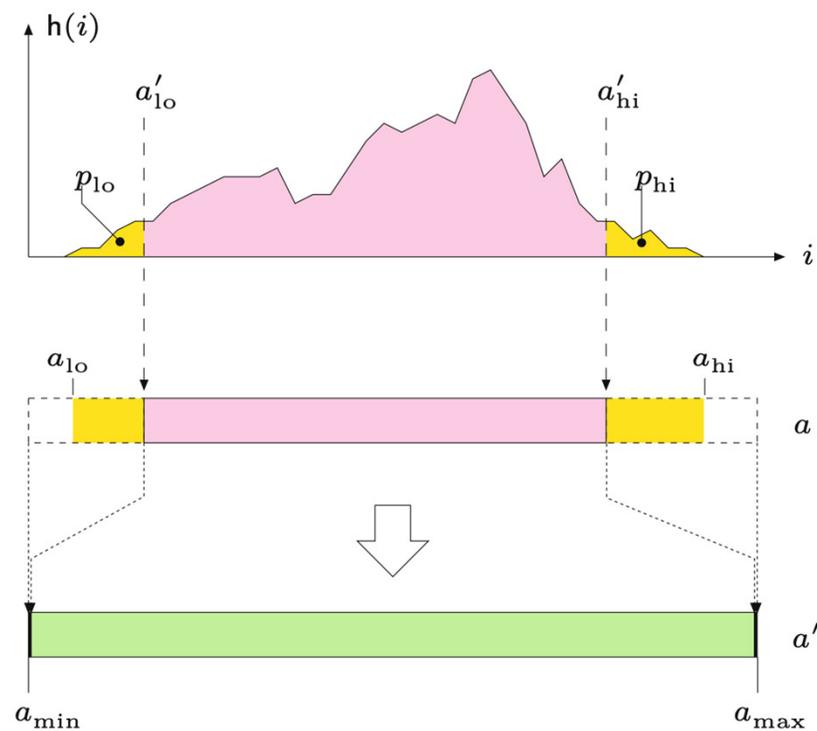
Autocontraste modificado

- En la práctica, el mapeo del autocontraste puede ser influenciado por valores de intensidad extremos.
- Esto se puede evitar *saturando* un porcentaje fijo (p_{lo} , p_{hi}) de pixeles en los extremos superior e inferior del rango de intensidad objetivo.
- Para lograr esto, se determinan dos valores límite a'_{lo} , a'_{hi} .
- Todos los pixeles con valor menor a a'_{lo} se mapean a a_{min} .
- Todos los pixeles con valor mayor a a'_{hi} se mapean a a_{max} .

Autocontraste modificado

Fig. 4.6

Modified auto-contrast operation (Eqn. (4.11)). Predefined quantiles (q_{l0} , q_{hi}) of image pixels—shown as dark areas at the left and right ends of the histogram $h(i)$ —are “saturated” (i.e., mapped to the extreme values of the target range). The intermediate values ($a = a'_{l0}, \dots, a'_{hi}$) are mapped linearly to the interval $a_{\min}, \dots, a_{\max}$.



Autocontraste modificado

- La función de mapeo para la operación de autocontraste modificado es:

$$f_{mac}(a) = \begin{cases} a_{min} & \text{para } a \leq a'_{lo} \\ a_{min} + (a - a'_{lo}) \cdot \frac{a_{max} - a_{min}}{a'_{hi} - a'_{lo}} & \text{para } a'_{lo} < a < a'_{hi} \\ a_{max} & \text{para } a \geq a'_{hi} \end{cases}$$

- Por lo general, se usa el mismo valor para los cuartiles superior e inferior (es decir, $p_{lo} = p_{hi} = p$), siendo $p = 0.005, \dots, 0.015$ (0.5%, ..., 1.5%) valores comunes.

Autocontraste modificado

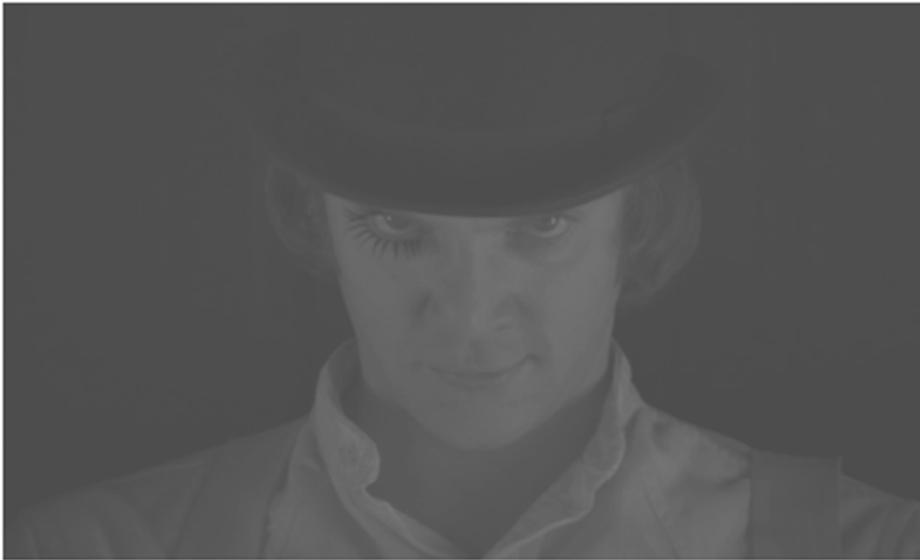
- Por ejemplo, la operación de autocontraste en Adobe Photoshop satura el 0.5% ($p = 0.005$) de todos los píxeles en ambos extremos del rango de intensidad.

Autocontraste en OpenCV

- OpenCV no tiene una función de autocontraste.
- Hay al menos 2 opciones:
 1. Utilizar la librería pillow.
 2. Implementar la función.
- En el código de la clase hay una implementación (`autocontrast_cv.py`).

Ejemplo

Original Image



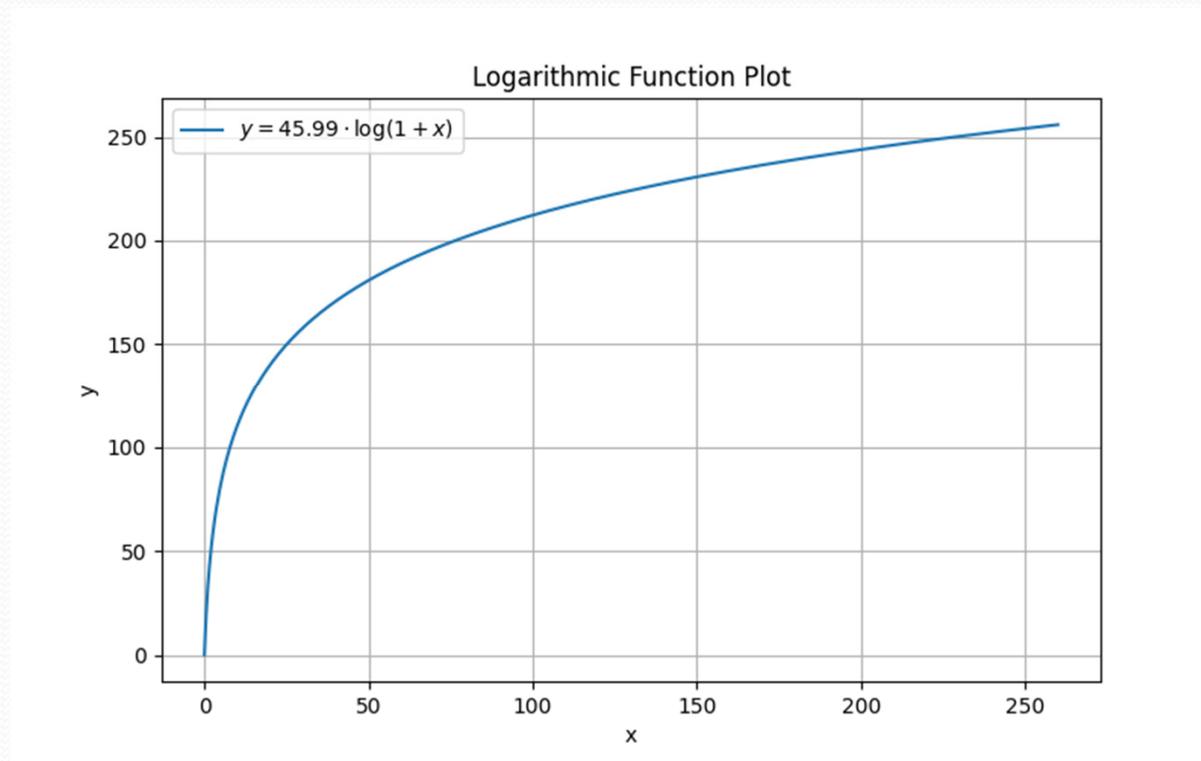
Autocontrast (cutoff=3)



Operador logaritmo

- Cada pixel se reemplaza por su logaritmo.
- El efecto es que se mejoran los valores de los pixeles de baja intensidad.
- La función es:
- $Q(i, j) = c \log(1 + |P(i, j)|)$
- c es el factor de escala para que el valor máximo de los pixeles en la imagen valga la intensidad máxima (255 en una imagen de 8 bits).
- $c = \frac{255}{\log(1+\max(I))}$

Operador logaritmo



Ejemplo

- Se puede usar para aclarar zonas oscuras.

Original Image



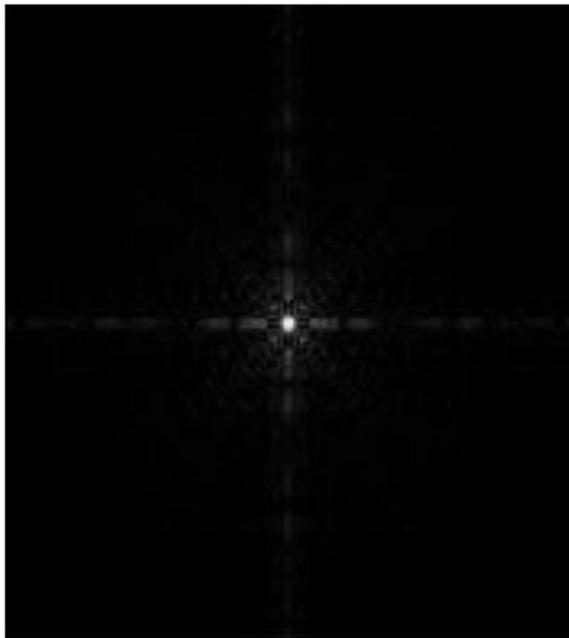
Logarithm Transformed Image



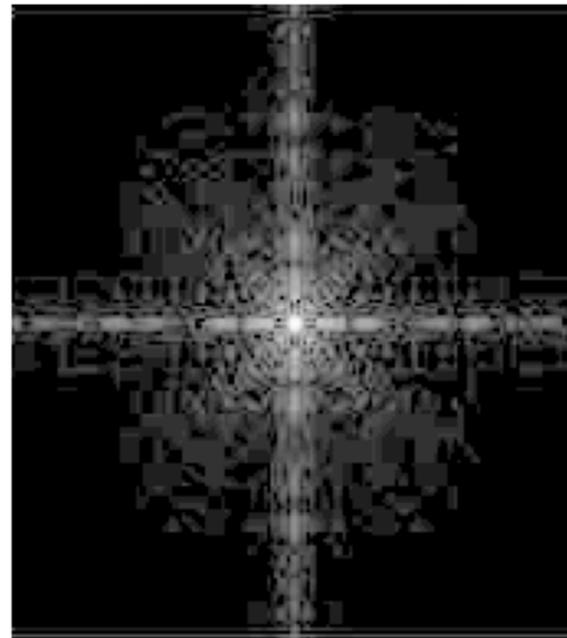
Ejemplo

- Se puede utilizar para revelar detalles ocultos.

Original Image



Logarithm Transformed Image



Operador logaritmo en OpenCV

- No está implementado en OpenCV.

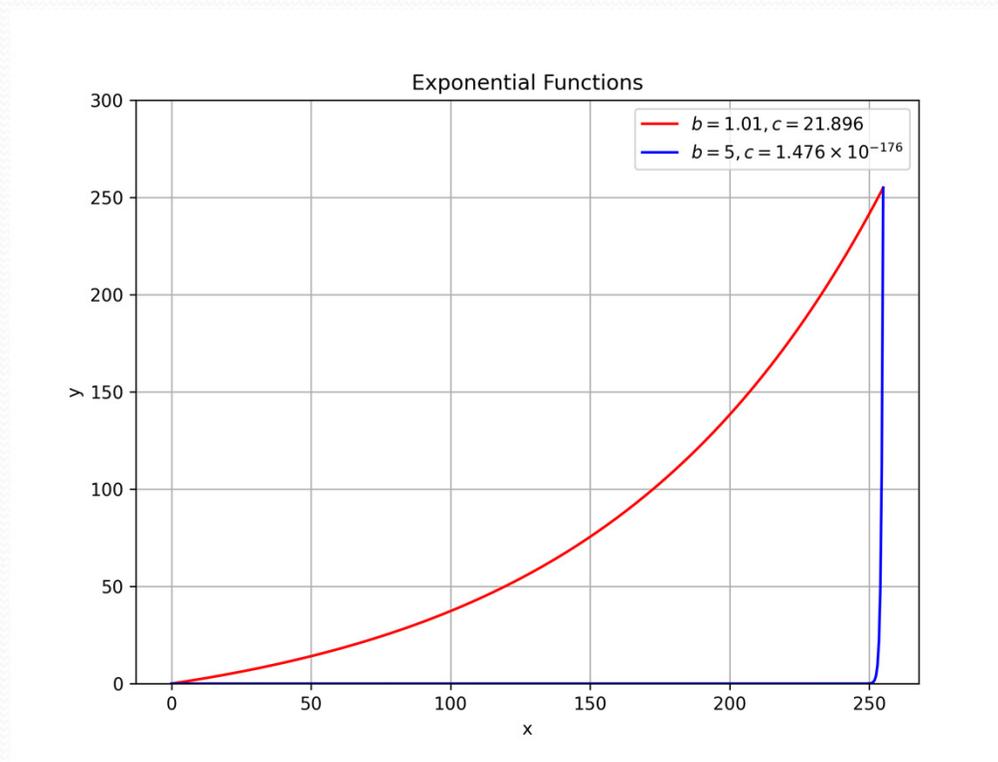
Operador exponencial

- Cada pixel se reemplaza por el valor de una base elevada al valor original del pixel.
- Es el inverso del operador logaritmo.
- El efecto es realzar los pixeles con intensidades altas.
- La función es:
- $Q(i, j) = c(b^{P(i, j)} - 1)$
- b es la base y c es un factor de escala para asegurar que el valor máximo de $Q(i, j)$ sea 255.

Operador exponencial

- $c = \frac{255}{b^{255}-1}$
- Por lo general, b es un valor ligeramente mayor que 1.

Operador exponencial



Ejemplo

- Revertir el operador logaritmo.

Original Image



Exponential (b=1.02)





Operador exponencial en OpenCV

- No está implementado en OpenCV.

Corrección gamma

- Cada pixel se reemplaza por su valor elevado a una potencia.
- $Q(i, j) = P(i, j)^\gamma$
- Donde γ es un valor real positivo.
- Si $P(i, j)$ se normaliza al intervalo $[0, 1]$, entonces $Q(i, j)$ también se queda dentro de $[0, 1]$.

Corrección gamma

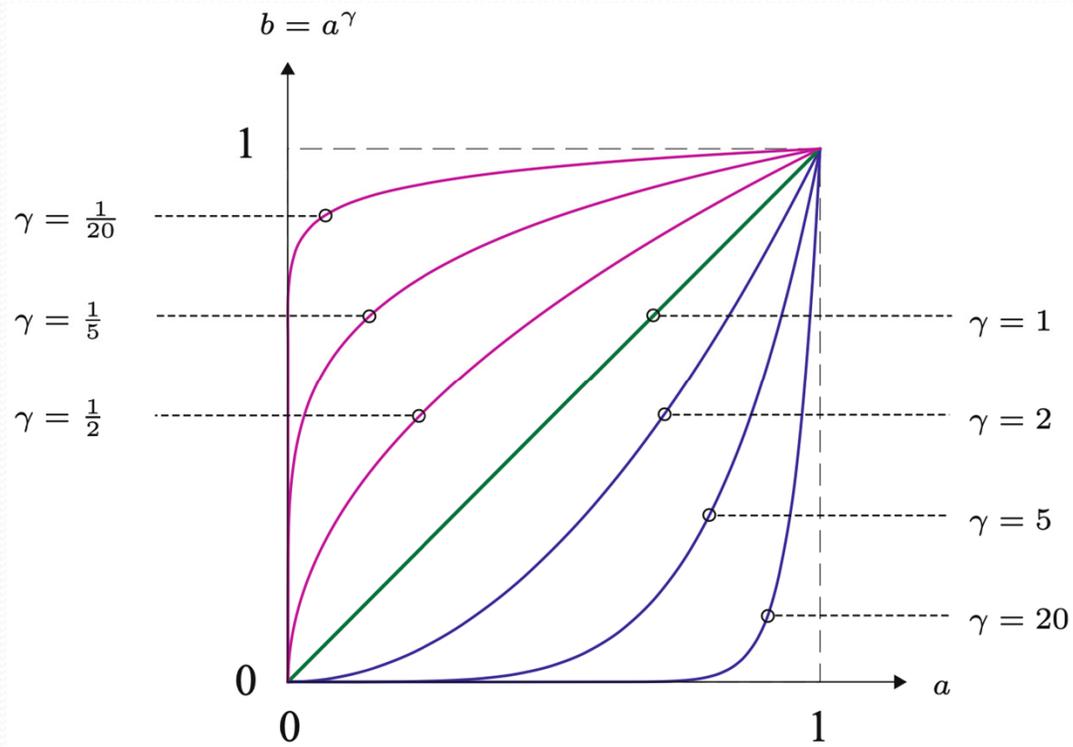


Fig. 4.18

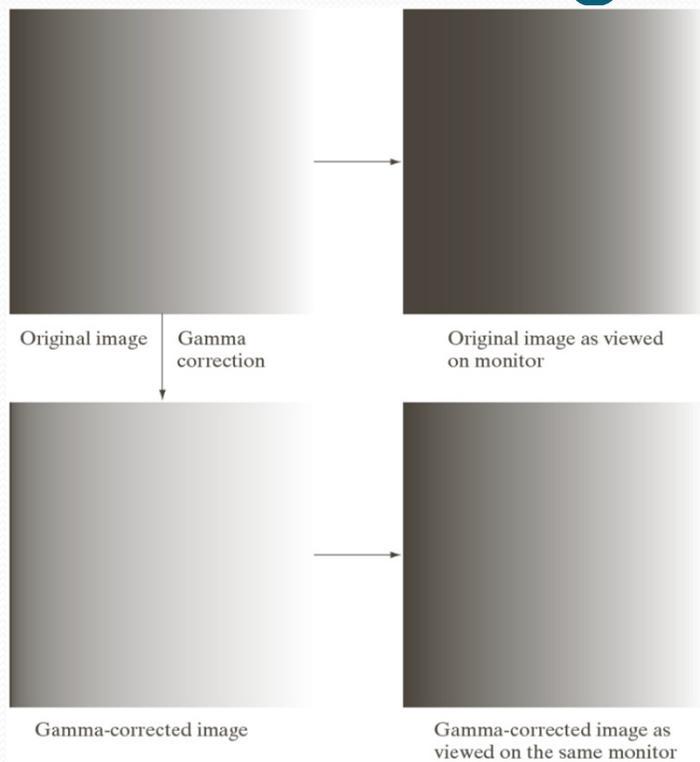
Gamma correction function $f_\gamma(a) = a^\gamma$ for $a \in [0, 1]$ and different gamma values.

Corrección gamma



Fuente: https://upload.wikimedia.org/wikipedia/commons/9/96/GammaCorrection_demo.jpg

Corrección gamma



a	b
c	d

FIGURE 3.7

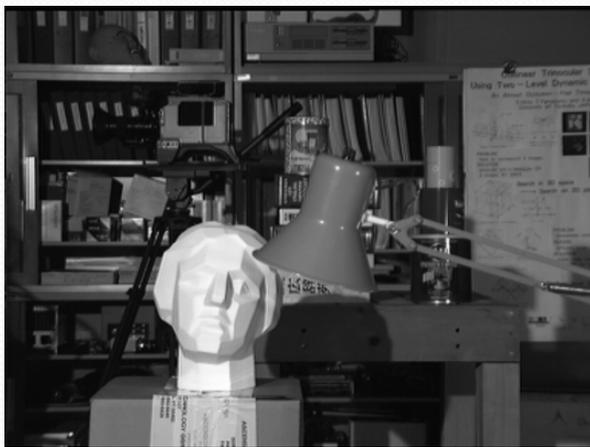
(a) Intensity ramp image. (b) Image as viewed on a simulated monitor with a gamma of 2.5. (c) Gamma-corrected image. (d) Corrected image as viewed on the same monitor. Compare (d) and (a).

Corrección gamma de (a) a (c):

$$s = r^{1/2.5} = r^{0.4}$$

Corrección gamma

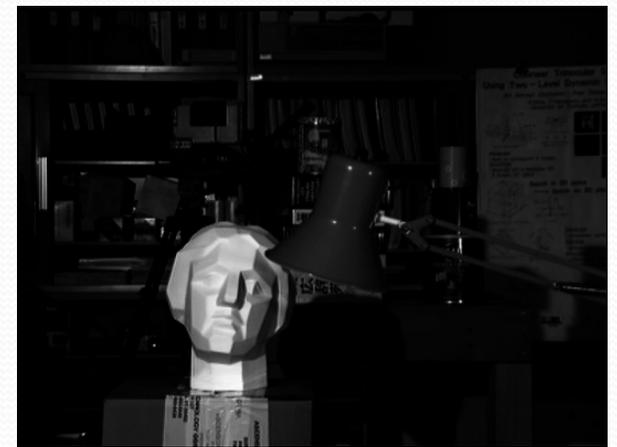
Original



$\gamma = 0.5$



$\gamma = 2.8$



Corrección gamma en OpenCV

- La corrección gamma no está implementada en OpenCV.