PROCESAMIENTO DE IMÁGENES DIGITALES

DPTO. MATEMÁTICA APLICADA I

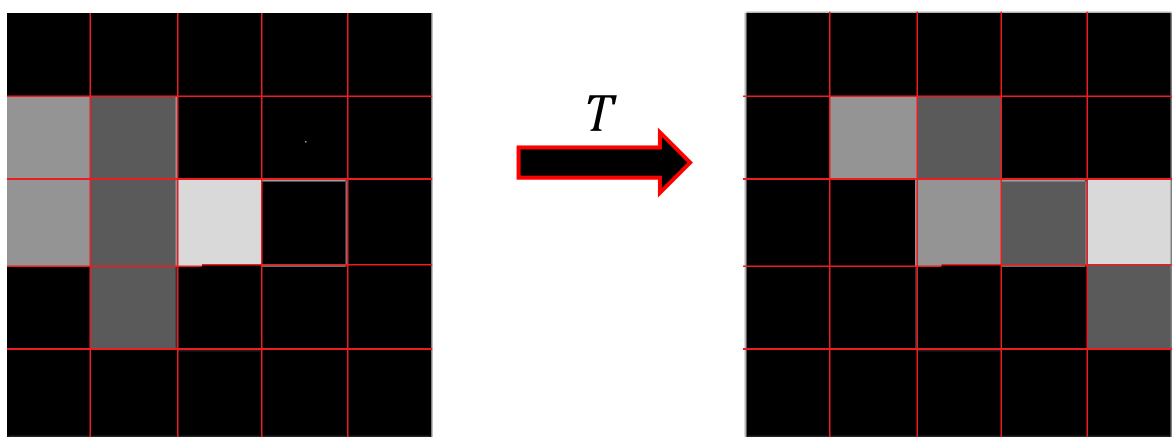
TRANSFORMACIONES GEOMÉTRICAS

Transformaciones geométricas

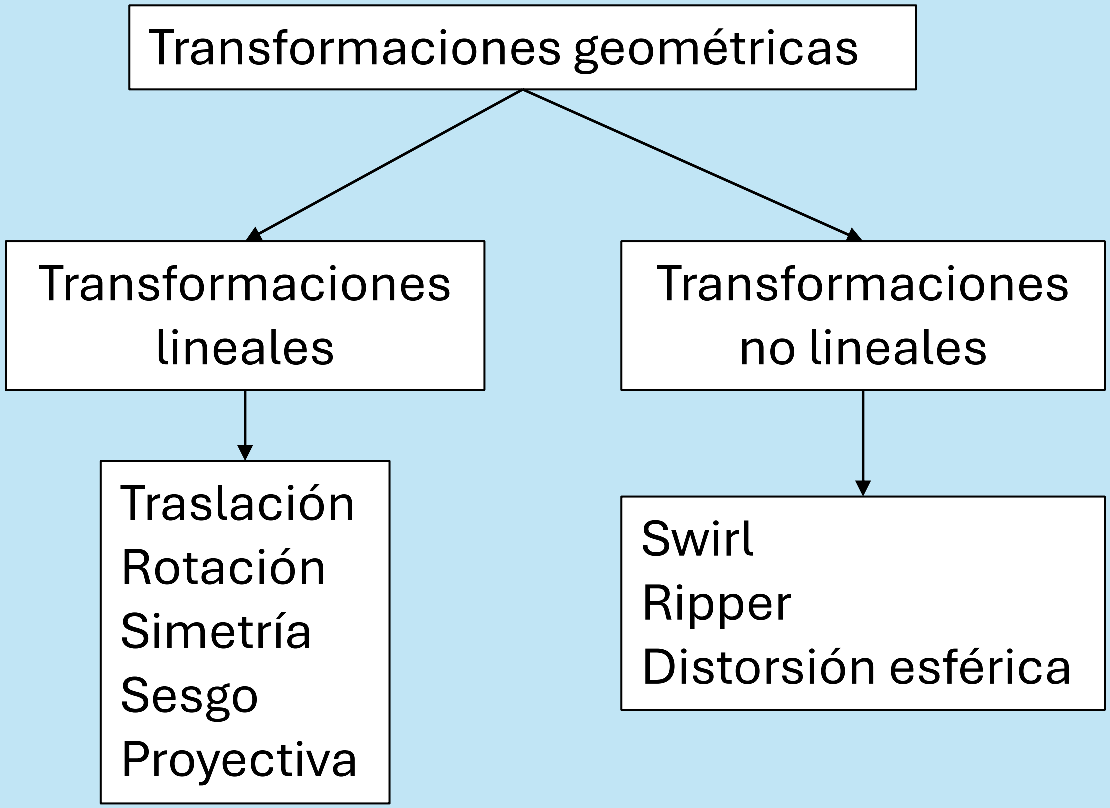
- **<u>Transformación geométrica</u>:** función T que modifica la relación espacial entre los píxeles de una imagen:

$$T(x, y) = (x', y')$$

donde (x, y) son las coordenadas de la imagen original y (x', y') las coordenadas de la imagen transformada.



Transformaciones geométricas



- **<u>Transformaciones afines</u>:**

Transformación geométrica en la que las coordenadas de la imagen transformada son expresadas como combinación lineal de las coordenadas de la imagen inicial más un término independiente.

$$\begin{cases} x' = ax + by + m \\ y' = cx + dy + n \end{cases} \rightarrow \begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} m \\ n \end{pmatrix} \rightarrow$$

Coordenadas homogéneas $\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} a & b & m \\ c & d & n \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$ $\rightarrow$ 6 grados de libertad

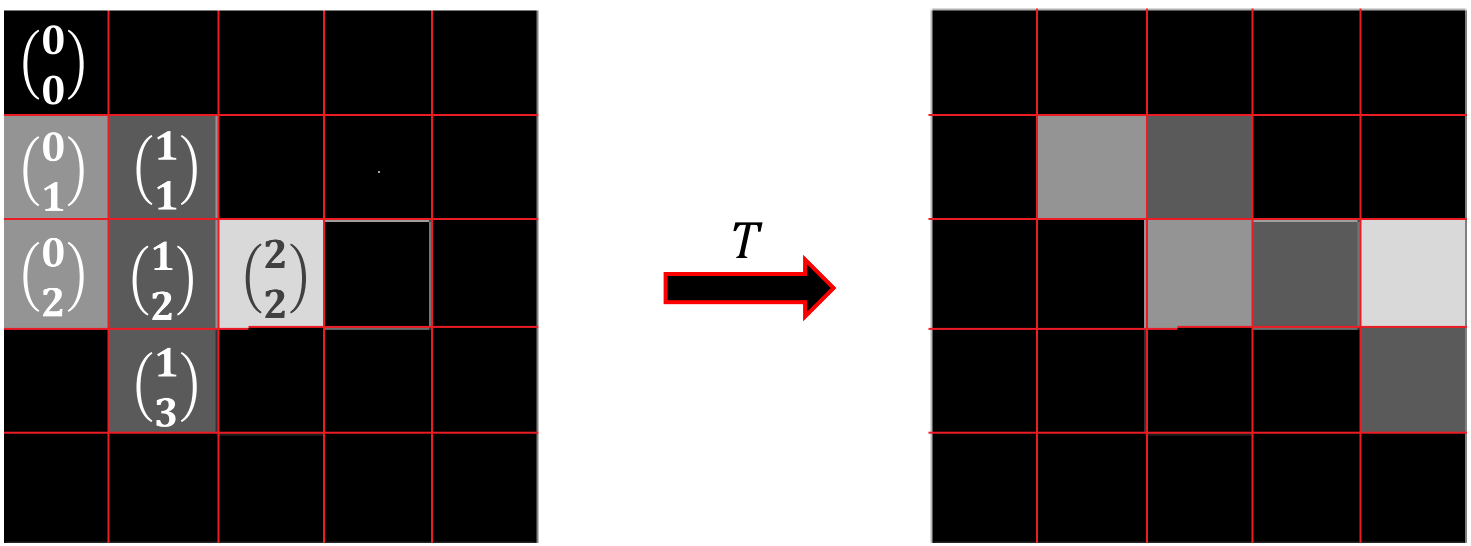
Al expresarse en forma matricial, si dicha matriz es no singular, se puede definir fácilmente la **inversa de la transformación.**

Transformaciones geométricas

- **<u>Transformaciones afines</u>:**

  <u>Ejemplo</u>: Sea T la transformación dada por:

$$T\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \end{pmatrix} \rightarrow \begin{cases} x' = x + y \\ y' = y \end{cases}$$



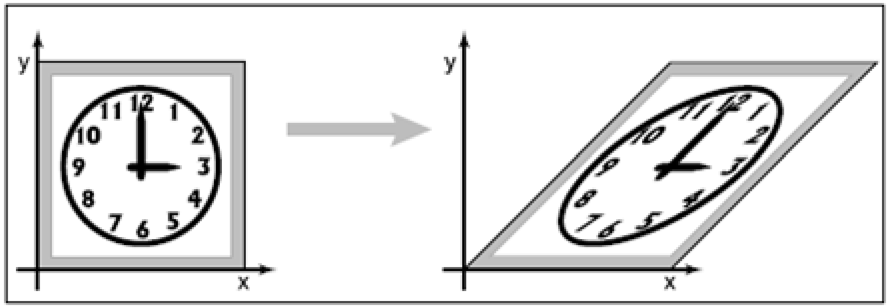
Transformaciones geométricas

- **<u>Transformaciones afines</u>:**

  <u>Ejemplo</u>: Sea T la transformación dada por:

$$T\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \end{pmatrix} \to \begin{cases} x' = x + y \\ y' = y \end{cases}$$

  El efecto de los coeficientes de la matriz de la transformación consiste en un empuje en dirección horizontal.



$$\begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

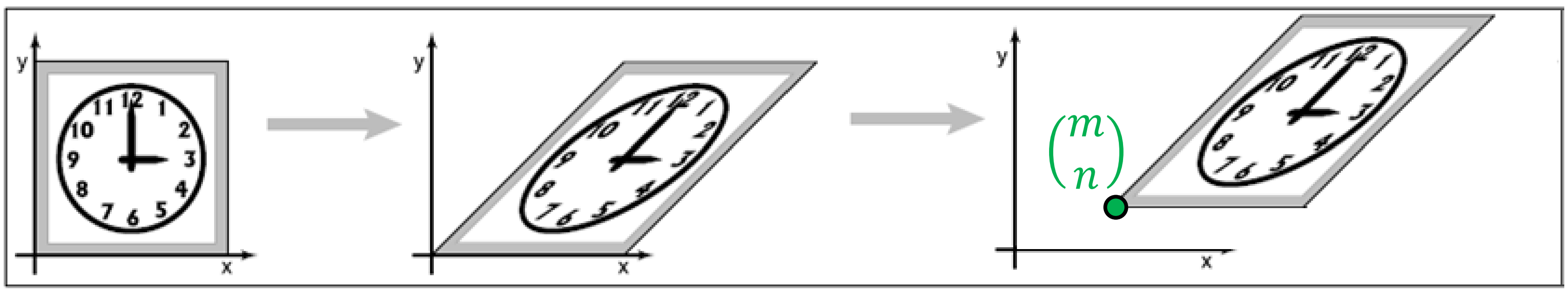
Transformaciones geométricas

- **<u>Transformaciones afines</u>:**

<u>Ejemplo</u>: Sea T la transformación dada por:

$$T\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} m \\ n \end{pmatrix} \rightarrow \begin{cases} x' = x + y + m \\ y' = y + n \end{cases}$$

El efecto de los coeficientes de la matriz de la transformación consiste en un empuje en dirección horizontal.



Transformaciones geométricas

- Clasificación: En función de las propiedades que preserva, una transformación geométrica se puede clasificar en:

    - Rígida: preserva distancias
    - Semejanza: preserva ángulos
    - Afinidad: preserva paralelismos
    - Proyectiva: preserva colinearidad

Transformaciones geométricas

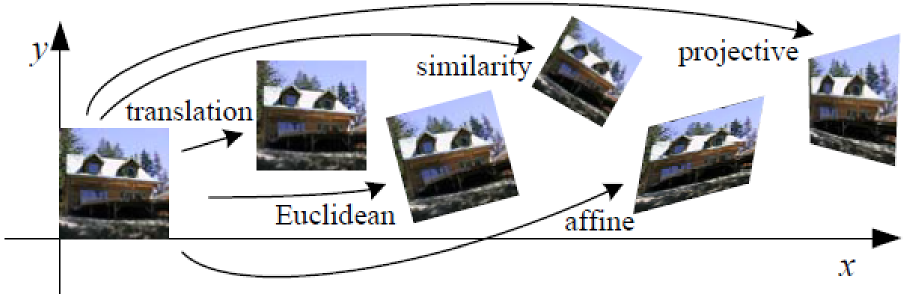


Figure 3.44 Basic set of 2D geometric image transformations.

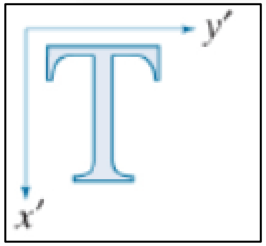
Transformation	Matrix	# DoF	Preserves	Icon
translation	$\begin{bmatrix} \mathbf{I} & \mathbf{t} \end{bmatrix}_{2\times 3}$	2	orientation	
rigid (Euclidean)	$\begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix}_{2\times 3}$	3	lengths	
similarity	$\begin{bmatrix} s\mathbf{R} & \mathbf{t} \end{bmatrix}_{2\times 3}$	4	angles	
affine	$\begin{bmatrix} \mathbf{A} \end{bmatrix}_{2\times 3}$	6	parallelism	
projective	$\begin{bmatrix} \tilde{\mathbf{H}} \end{bmatrix}_{3\times 3}$	8	straight lines	

Table 3.3 Hierarchy of 2D coordinate transformations. Each transformation also preserves the properties listed in the rows below it, i.e., similarity preserves not only angles but also parallelism and straight lines. The 2 × 3 matrices are extended with a third $[\mathbf{0}^T\ 1]$ row to form a full 3 × 3 matrix for homogeneous coordinate transformations.

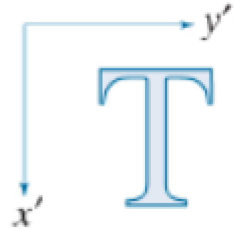
Transformaciones geométricas

- Transformaciones geométricas
  - Transformaciones lineales
    - Traslación
    - Rotación
    - Simetría
    - Sesgo
    - Proyectiva
  - Transformaciones no lineales
    - Swirl
    - Ripper
    - Distorsión esférica

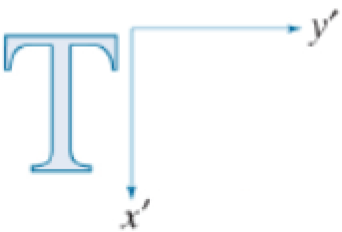
Transformaciones geométricas

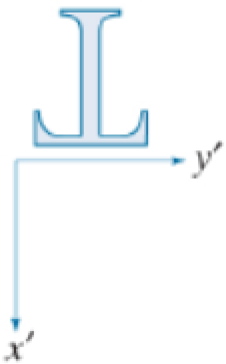


- <u>Traslación</u> → $M = \begin{pmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{pmatrix}$



(Rígida)

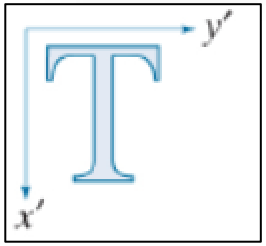




- <u>Simetría/reflexión</u> → $M = \begin{pmatrix} \pm 1 & 0 & 0 \\ 0 & \pm 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$

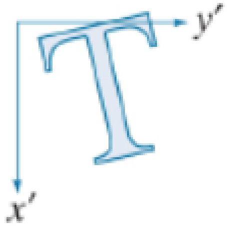
(Rígida)

Transformaciones geométricas



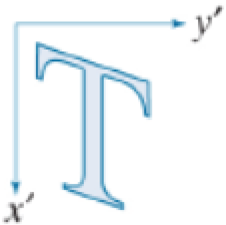
- <u>Rotación</u> $\rightarrow M = \begin{pmatrix} \cos(\theta) & -sen(\theta) & 0 \\ sen(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix}$

(Rígida)

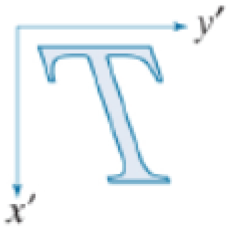


- <u>Sesgo</u> $\rightarrow M = \begin{pmatrix} 1 & s_v & 0 \\ s_h & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$

(Afín)



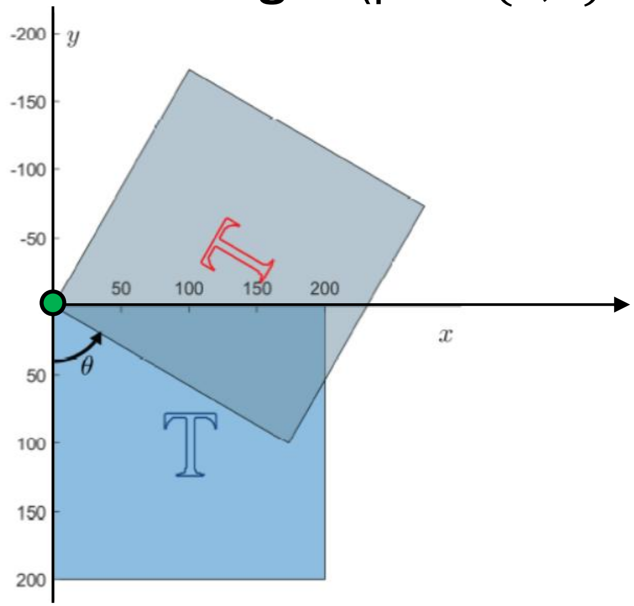
sesgo vertical
$s_v = 1, s_h = 0$



sesgo horizontal
$s_v = 0, s_h = 1$

Transformaciones geométricas

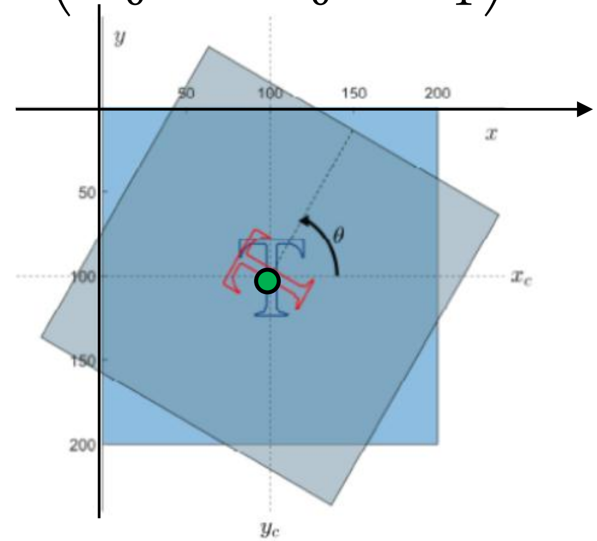
- Las matrices anteriores se refieren a transformaciones que dejan **invariante el origen** (pixel (0,0) en el caso de imágenes).



$$\begin{pmatrix} \cos(\theta) & -sen(\theta) & 0 \\ sen(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

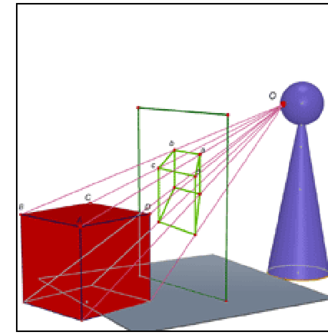
Rotación respecto al origen.

$$\begin{pmatrix} \cos(\theta) & -sen(\theta) & x_c \\ sen(\theta) & \cos(\theta) & y_c \\ 0 & 0 & 1 \end{pmatrix}$$



Rotación respecto al centro geométrico de la imagen (x_c, y_c).

Transformaciones geométricas



- Transformación proyectiva → $M = \begin{pmatrix} a & b & c \\ d & e & f \\ g & h & 1 \end{pmatrix}$

8 grados de libertad

Practica:

- Visita la página del tutorial de OpenCV:

https://docs.opencv.org/4.x/da/d6e/tutorial_py_geometric_transformations.html

- Averigua qué funciones tiene implementadas OpenCV para realizar las transformaciones vistas.
- Averigua qué funciones ofrece OpenCV para obtener la matriz de la transformación deseada.

Transformaciones geométricas

- Transformaciones geométricas
  - Transformaciones lineales
    - Traslación
    - Rotación
    - Simetría
    - Sesgo
    - Proyectiva
  - Transformaciones no lineales
    - Swirl
    - Ripper
    - Distorsión esférica

Tema 6: Transformaciones geométricas

- Transformación swirl: Produce una rotación en torno a un punto (x_c, y_c) que decrece conforma el radio aumenta.

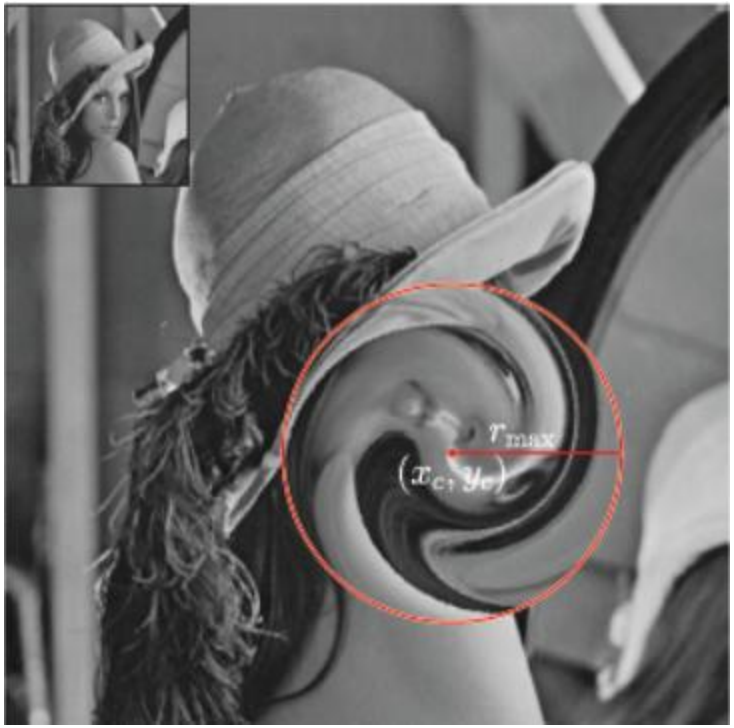
$$\hat{x} = \begin{cases} x_c + r \cdot \cos\beta & r \leq r_{\text{máx}} \\ x & r > r_{\text{máx}} \end{cases}$$

$$\hat{y} = \begin{cases} y_c + r \cdot \sin\beta & r \leq r_{\text{máx}} \\ y & r > r_{\text{máx}} \end{cases}$$

$$r = \sqrt{d_x^2 + d_y^2}$$

$$\beta = \operatorname{atan}\left(\frac{d_y}{d_x}\right) + \alpha\left(\frac{r_{\text{máx}} - r}{r_{\text{máx}}}\right)$$

donde $d_x = x - x_c$ y $d_y = y - y_c$.



Transformaciones geométricas

- Transformación ripper: Produce una distorsión local en forma de onda sinusoidal en la dirección x ó y.

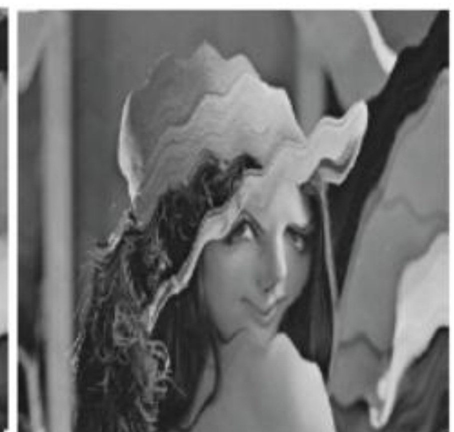
$$\begin{aligned}\hat{x} &= x + a_x \cdot \sin\left(\frac{2\pi y}{\tau_x}\right)\\ \hat{y} &= y + a_y \cdot \sin\left(\frac{2\pi x}{\tau_y}\right)\end{aligned}$$

τ_x y τ_y → periodos de la longitud de ondas

a_x y a_y → amplitudes de los desplazamientos

Imagen original

Ripper en x



Ripper en y

Transformaciones geométricas

- <u>Transformación distorsión esférica</u>: Simula el efecto de una lente esférica cuyos parámetros son su centro (x_c, y_c), radio de distorsión r_{max} e índice de la lente ρ.

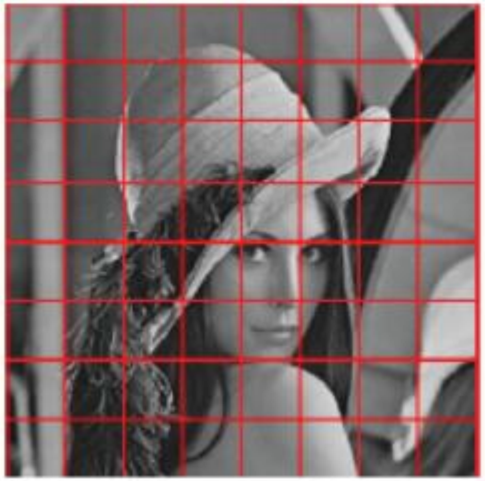
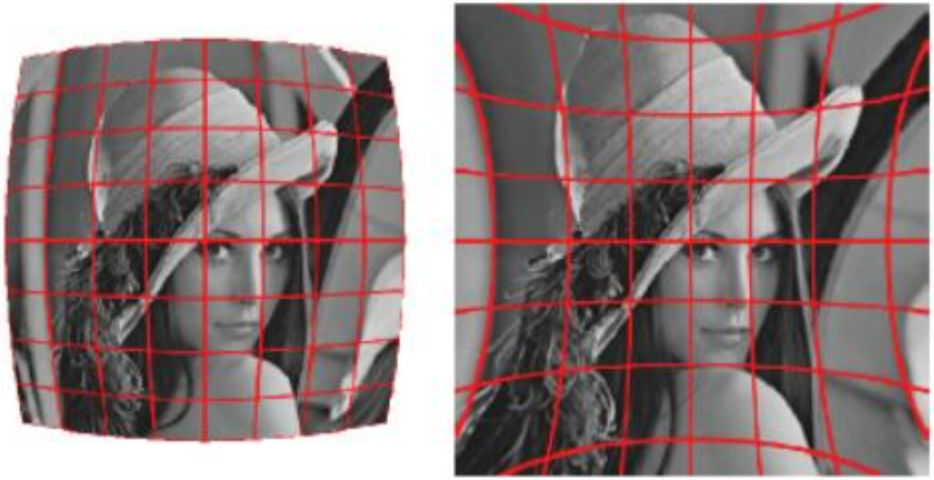


Imagen original con cuadrícula de referencia



Dos tipos de distorsiones

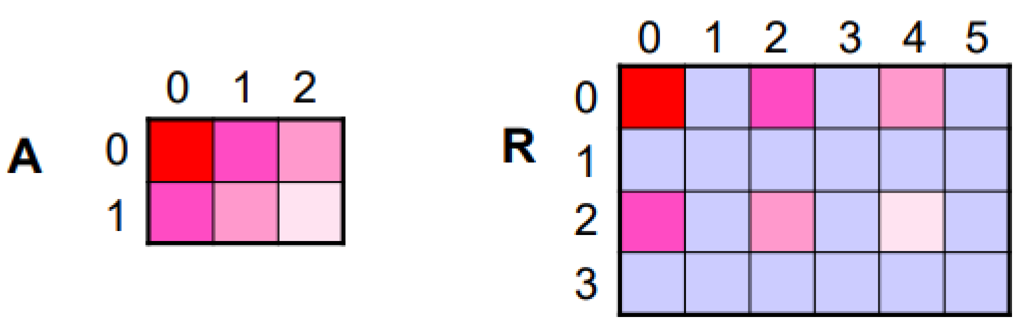
Transformaciones geométricas

- ¿Qué ocurre si el resultado de aplicar una transformación a un pixel (x, y) es un número no entero?

- Por ejemplo, supongamos que queremos **aumentar la dimensión de una imagen** al doble. La matriz de la transformación sería:

$$M = \begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

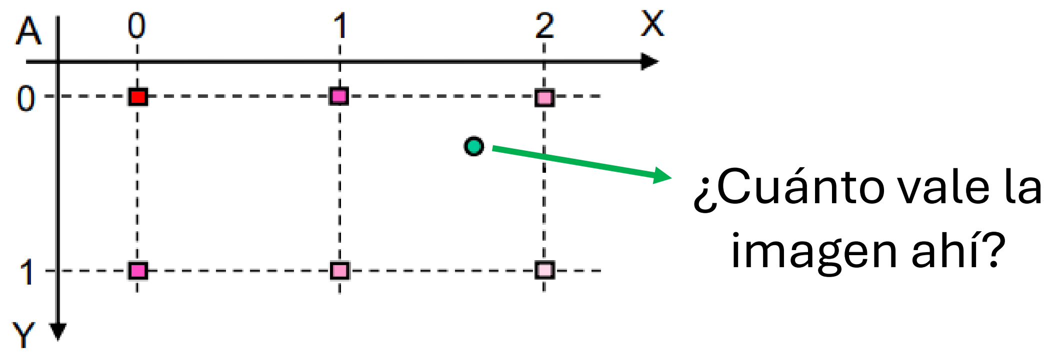
¡Índices no definidos en la matriz!



- $R(0,0) = A(0,0)$
- $R(1,0) = A(0.5,0)$
- $R(1,1) = A(0.5,0.5)$

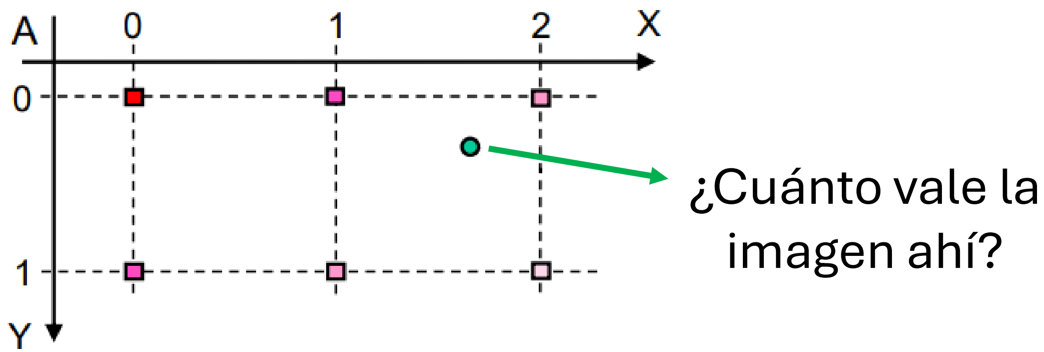
Transformaciones geométricas

- **Problema:** las imágenes son señales discretas, pero la transformación geométrica las trata como si fueran continuas (definidas en todo el plano).



Transformaciones geométricas

- **Problema:** las imágenes son señales discretas, pero la transformación geométrica las trata como si fueran continuas (definidas en todo el plano).



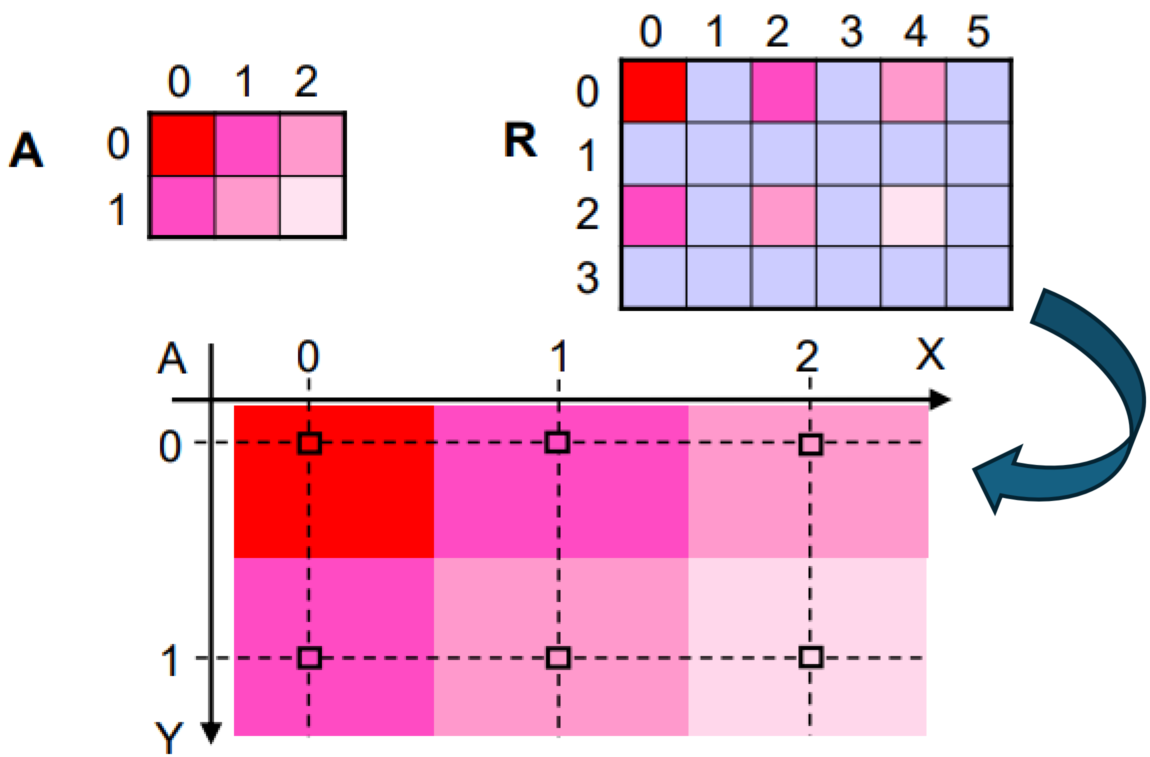
Interpolación

vecino más próximo | bilineal | bicúbica

Transformaciones geométricas

- **<u>Interpolación vecino más próximo</u>:**

  Cualquier punto del espacio toma el valor del píxel más cercano.



- **Interpolación vecino más próximo:**

Ejemplo: Zoom 10 veces mayor usando el vecino más próximo.

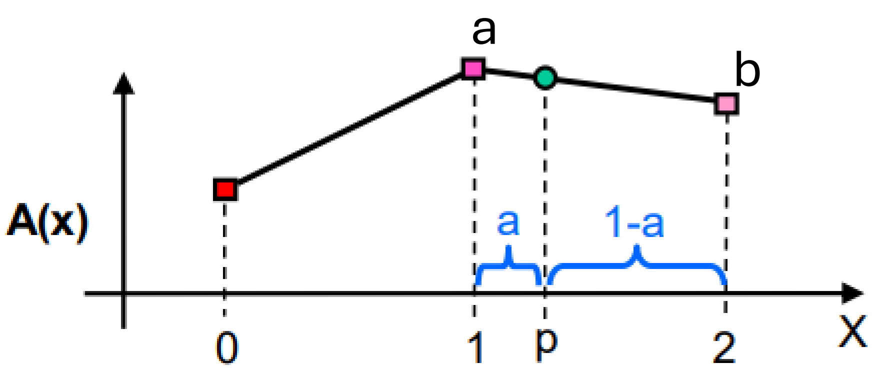
Imagen original
25×26

Imagen ampliada
250×260

- **Ventajas**: Es muy sencillo y rápido de calcular.
- **Inconvenientes**: El efecto de cuadriculado es evidente, y da lugar imágenes de poca calidad.

- **<u>Interpolación bilineal</u>:**

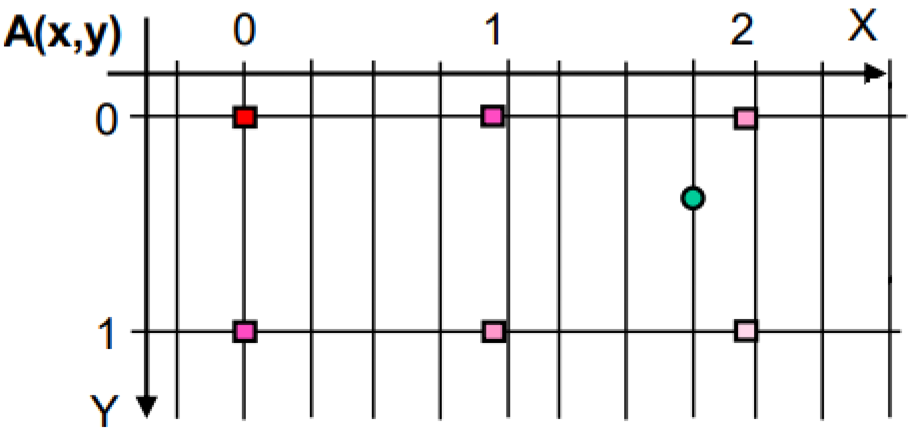
  En una dimensión, una interpolación lineal significa trazar una línea recta entre cada par de puntos consecutivos.

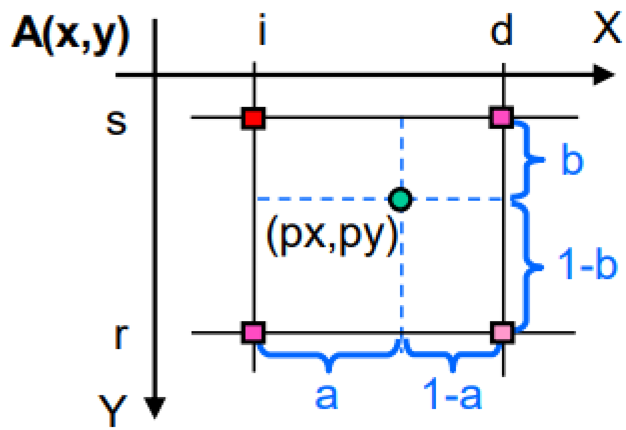


$$A'(p) = (1 - a)A(a) + aA(b)$$

- **<u>Interpolación bilineal</u>:**

  En dos dimensiones, la interpolación bilineal consiste en aplicar dos interpolaciones lineales: una horizontal y una vertical.





$$A'(p_x, s) = (1-a)A(i,s) + a\,A(d,s)$$
$$A'(p_x, r) = (1-a)A(i,r) + a\,A(d,r)$$

$$\Longrightarrow \quad A'\left(p_x, p_y\right) = (1-b)A'(p_x, s) + bA'(p_x, r)$$

Transformaciones geométricas

- **Interpolación bilineal:**

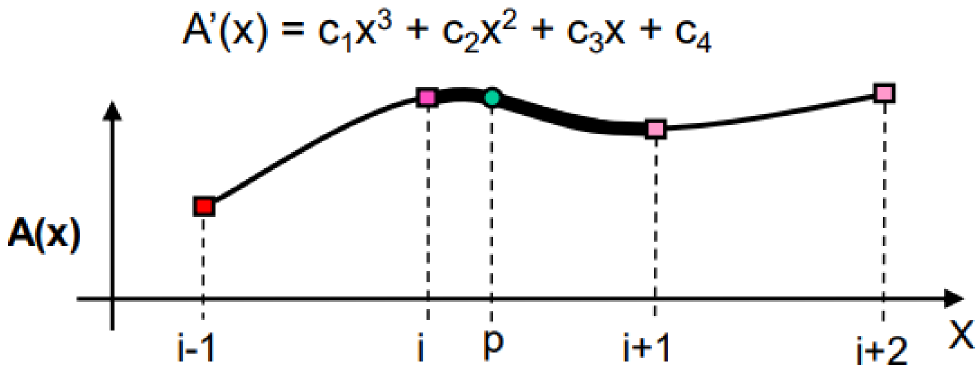
  Ejemplo: Zoom 10 veces mayor usando interpolación bilineal.



Observación: un zoom entero de K con interpolación bilineal es parecido (= a veces) a un zoom de K con vecino más próximo, seguido de un filtro de media de $\mathrm{K} \times \mathrm{K}$.

- **Interpolación bicúbica:**

En una dimensión, la interpolación cúbica consiste en trazar una cúbica entre los 4 puntos más próximos (2 a la izquierda y 2 a la derecha).



p punto a interpolar
Obtener las 4 ecuaciones:
$A'(i-1) = A(i-1), A'(i) = A(i), A'(i+1) = A(i+1), A'(i+2) = A(i+2)$
4 ecuaciones, 4 incógnitas: se obtienen los coeficientes c_1, c_2, c_3, c_4

Transformaciones geométricas

- **<u>Interpolación bicúbica</u>:**

  En dos dimensiones, la interpolación bicúbica consiste en aplicar dos interpolaciones bilineales: una horizontal y una vertical.

  <u>Ejemplo</u>: Zoom 10 veces mayor usando interpolación bicúbica.

Transformaciones geométricas

- **Interpolación**

Practica:

- Usa la función de OpenCV resize() para aumentar el tamaño de una imagen.
- Averigua qué opciones ofrece de interpolación, como argumentos de la función.

Transformaciones geométricas

- **Bibliografía y recursos:**

- Szeliski, Richard. *Computer vision: algorithms and applications*. Springer Nature, 2022.

- OpenCV tutorial: Geometric transformations of images. https://docs.opencv.org/4.x/da/d6e/tutorial_py_geometric_transformations.html (visitado el 4/10/2024)