DEPARTAMENTO DE
INGENIERÍA
INFORMÁTICA
UNIVERSIDAD DE SANTIAGO DE CHILE

Procesamiento y Análisis de Imágenes

Violeta Chang

violeta.chang@usach.cl

Créditos por slides: José M. Saavedra, Harvey Rhody, Emmanuel Agu

Transformaciones Geométricas

A spatial transformation of an image is a geometric transformation of the image coordinate system.

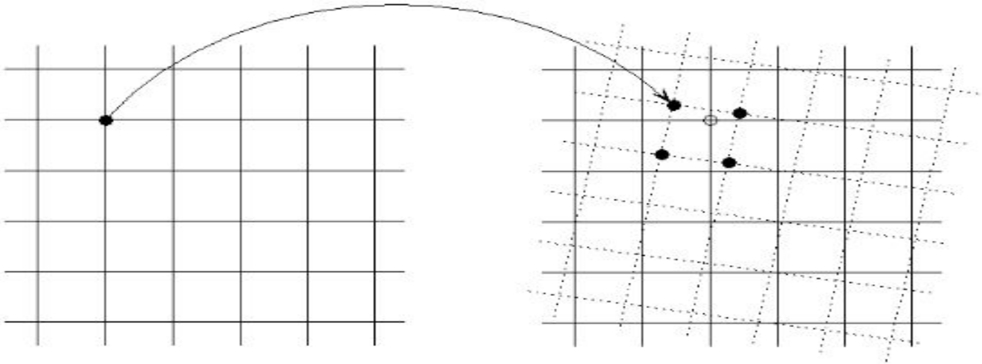
It is often necessary to perform a spatial transformation to:

- Align images that were taken at different times or with different sensors
- Correct images for lens distortion
- Correct effects of camera orientation
- Image morphing or other special effects

Transformaciones Geométricas

In a spatial transformation each point (x, y) of image A is mapped to a point (u, v) in a new coordinate system.

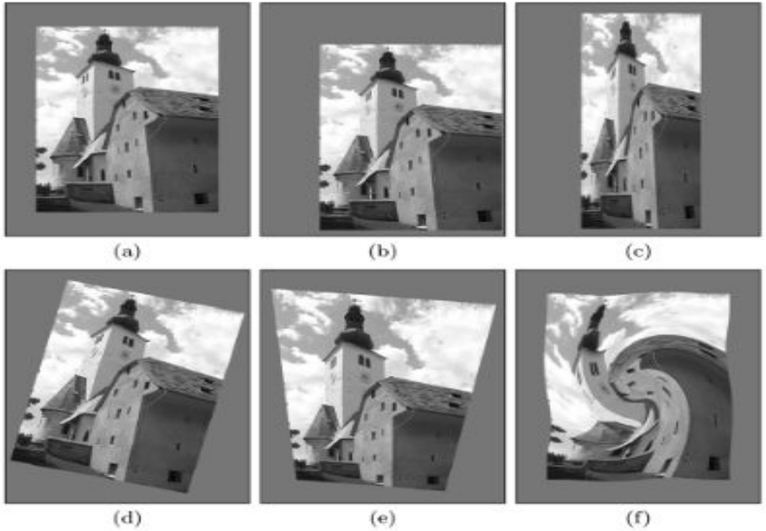
$$u = f_1(x, y)$$
$$v = f_2(x, y)$$



Mapping from (x, y) to (u, v) coordinates. A digital image array has an implicit grid that is mapped to discrete points in the new domain. These points may not fall on grid points in the new domain.

Transformaciones Geométricas

- Filters, point operations change intensity
- Pixel position (and geometry) unchanged
- Geometric operations: change image geometry
- **Examples:** translating, rotating, scaling an image



Examples of Geometric operations

Transformaciones Geométricas

- Example applications of geometric operations:
  - Zooming images, windows to arbitrary size
  - Computer graphics: deform textures and map to arbitrary surfaces
- **Definition:** Geometric operation transforms image *I* to new image *I'* by modifying **coordinates of image pixels**

$$I(x,y) \rightarrow I'(x',y')$$

- Intensity value originally at (x,y) moved to new position (x',y')

Example: Translation geometric operation moves value at (x,y) to (x + d_x, y + d_y)

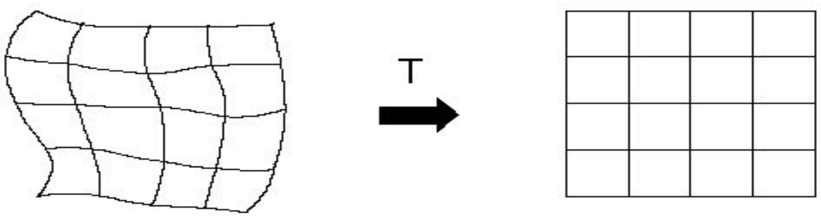
Transformaciones Geométricas

- Since image coordinates can only be discrete values, some transformations may yield (x',y') that's not discrete
- **Solution:** interpolate nearby values

Transformaciones Geométricas

- Modifican la relación espacial entre pixeles en una imagen.
- En términos de procesamiento de imágenes, una transformación geométrica consiste de:
- (1) TRANSFORMACIÓN DE COORDENADAS
- (2) INTERPOLACIÓN DE INTENSIDADES , que asigne un valor a las nuevas coordenadas.

TRANSFORMA EL DOMINIO



Transformaciones Geométricas

- **Translation:** (shift) by a vector (d_x, d_y)

$$T_x : x' = x + d_x \qquad \text{or} \qquad \begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} d_x \\ d_y \end{pmatrix}$$
$$T_y : y' = y + d_y$$

- **Scaling:** (contracting or stretching) along x or y axis by a factor s_x or s_y

$$T_x : x' = s_x \cdot x \qquad \text{or} \qquad \begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} s_x & 0 \\ 0 & s_y \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix}$$
$$T_y : y' = s_y \cdot y$$

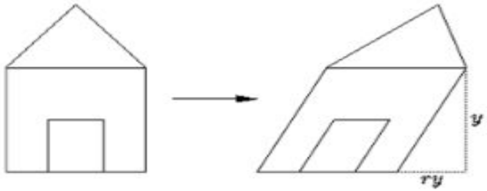
Transformaciones Geométricas

- **Shearing:** along x and y axis by factor b_x and b_y

$$T_x : x' = x + b_x \cdot y$$
$$T_y : y' = y + b_y \cdot x$$

or

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} 1 & b_x \\ b_y & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix}$$



- **Rotation:** the image by an angle α

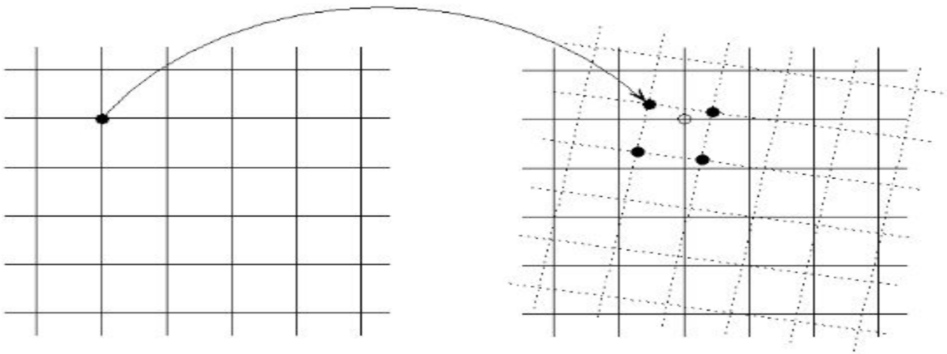
$$T_x : x' = x \cdot \cos\alpha - y \cdot \sin\alpha$$
$$T_y : y' = x \cdot \sin\alpha + y \cdot \cos\alpha$$

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix}$$

Transformaciones Geométricas

Interpolation

Interpolation is needed to find the value of the image at the grid points in the target coordinate system. The mapping T locates the grid points of A in the coordinate system of B, but those grid points are not on the grid of B.



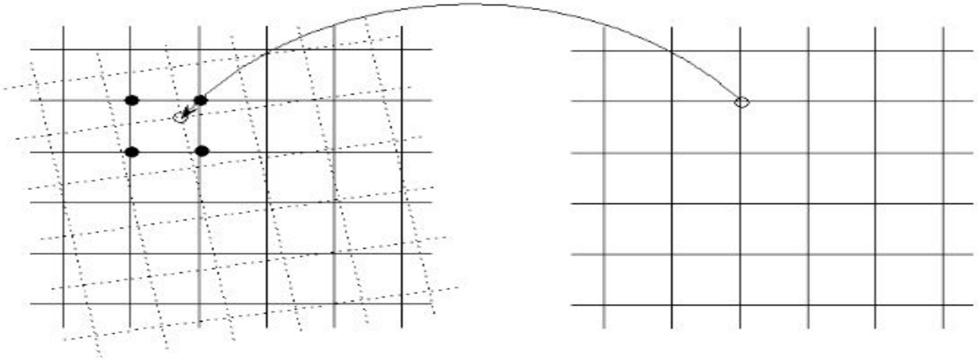
To find the values on the grid points of B we need to interpolate from the values at the projected locations.

Finding the closest projected points to a given grid point can be computationally expensive.

Transformaciones Geométricas

Inverse Projection

Projecting the grid of B into the coordinate system of A maintains the known image values on a regular grid. This makes it simple to find the nearest points for each interpolation calculation.



Let $\mathbf{Q}_g$ be the homogeneous grid coordinates of B and let $\mathbf{H}$ be the transformation from A to B. Then

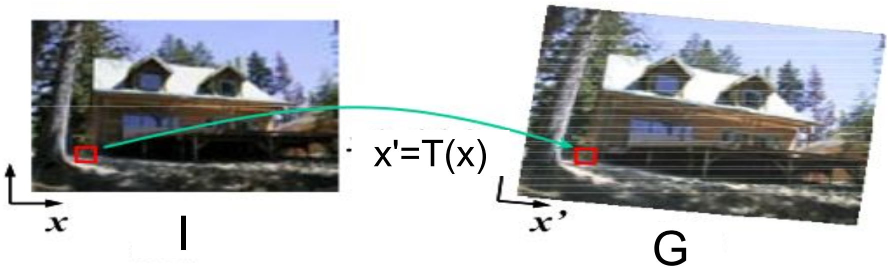
$$\mathbf{P} = \mathbf{H}^{-1}\mathbf{Q}_g$$

represents the projection from B to A. We want to find the value at each point $\mathbf{P}$ given from the values on $\mathbf{P}_g$, the homogeneous grid coordinates of A.

Transformaciones Geométricas

- **Interpolación de Intensidades**
  - Sea una imagen de entrada *I* y una transformación *T*, se deben calcular los valores para cada pixel $x'=T(x)$ en la imagen de salida *G*.

$$G(x')=?$$



Transformaciones Geométricas

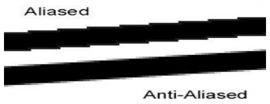
- **Interpolación de Intensidades**
  - Estrategia hacia adelante (forward mapping)

```
Algoritmo:  forward mapping
INPUT: I, T

Para cada pixel x en I(x)
        1) Calcular la ubicación destino x'=T(x)
        2) Copiar I(x) a G(x')
retornar G
```

Problemas

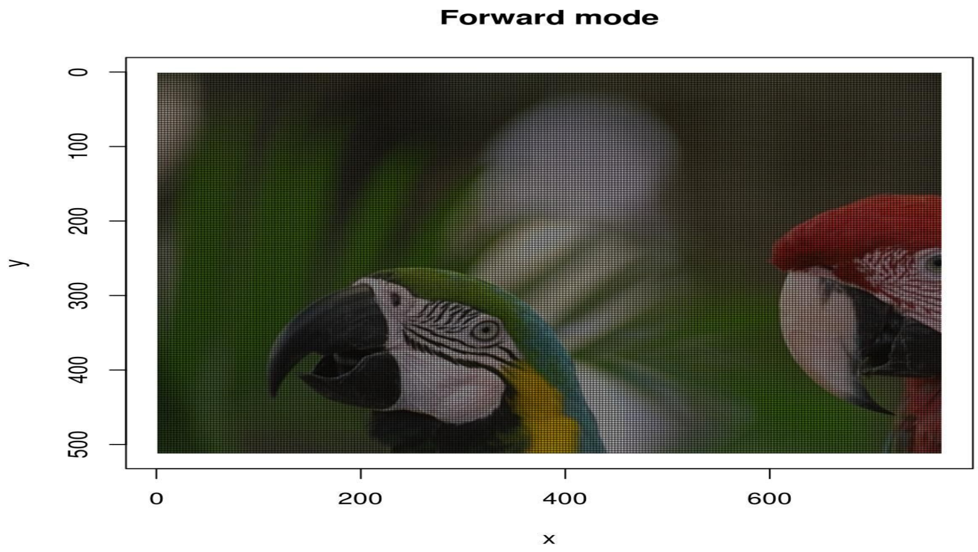
- Aliasing



- Dos puntos en *I* pueden estar asociados a un mismo punto *G.*
- Pueden quedar puntos sin valor en *G.*

Transformaciones Geométricas

- **Interpolación de Intensidades**
  - Estrategia hacia adelante (forward mapping)



Interpolación de intensidades

- **Interpolación de Intensidades**
  - Estrategia inversa (inverse mapping)

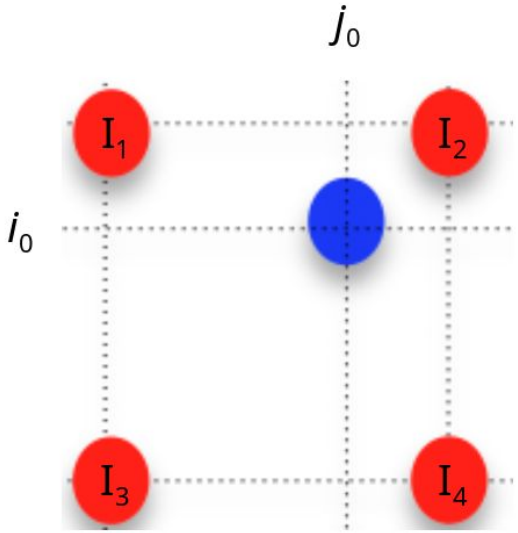
Algoritmo: inverse mapping
INPUT: I, T

Para cada pixel x' en $G(x')$
	1) Calcular la ubicación fuente $x=T^{-1}(x')$ usando la transformación inversa.
	2) Interpolar $I(x)$ y copiar el valor resultante en $G(x')$.
retornar G

Técnicas de interpolación:
- Vecino más cercano
- Interpolación bilineal
- Interpolación bicúbica

Interpolación de intensidades

- Interpolación de Intensidades
  - Vecino más cercano

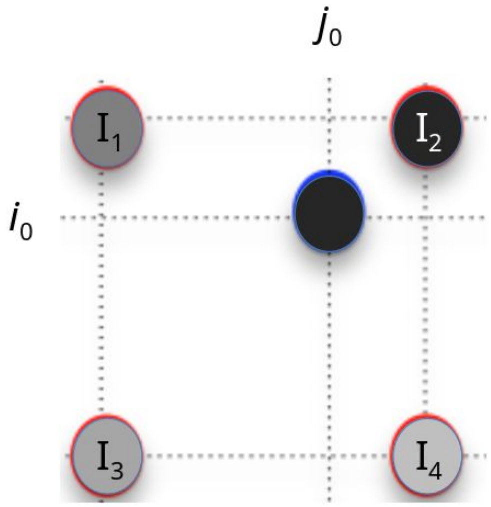


Nearest pixel

$J(i,j)$ = interpolation $\{I(i_0,j_0)\} = I_2$

Interpolación de intensidades

- Interpolación de Intensidades
  - Vecino más cercano

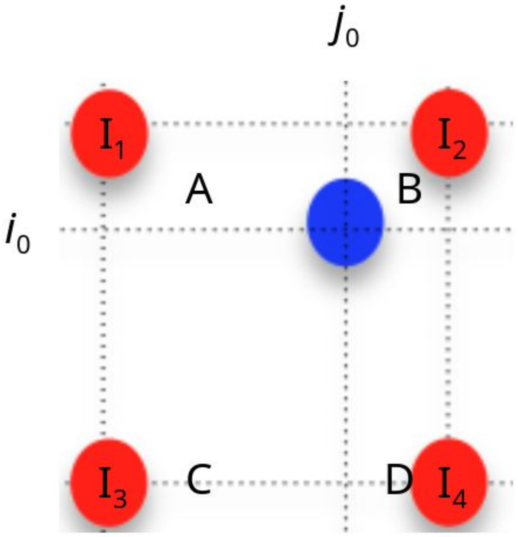


Nearest pixel

$J(i,j)$ = interpolation $\{I(i_0,j_0)\} = I_2$

Interpolación de intensidades

- Interpolación de Intensidades
  - Interpolación bilineal



Bilinear interpolation

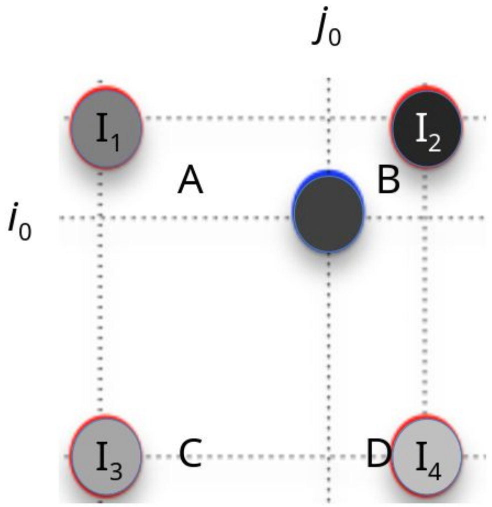
$J(i,j)$ = interpolation $\{I(i_0,j_0)\}$ =

$AI_4+BI_3+CI_2+DI_1$

$A+B+C+D = 1$

Interpolación de intensidades

- Interpolación de Intensidades
  - Interpolación bilineal



Bilinear interpolation

$J(i,j)$ = interpolation $\{I(i_0,j_0)\}$ =

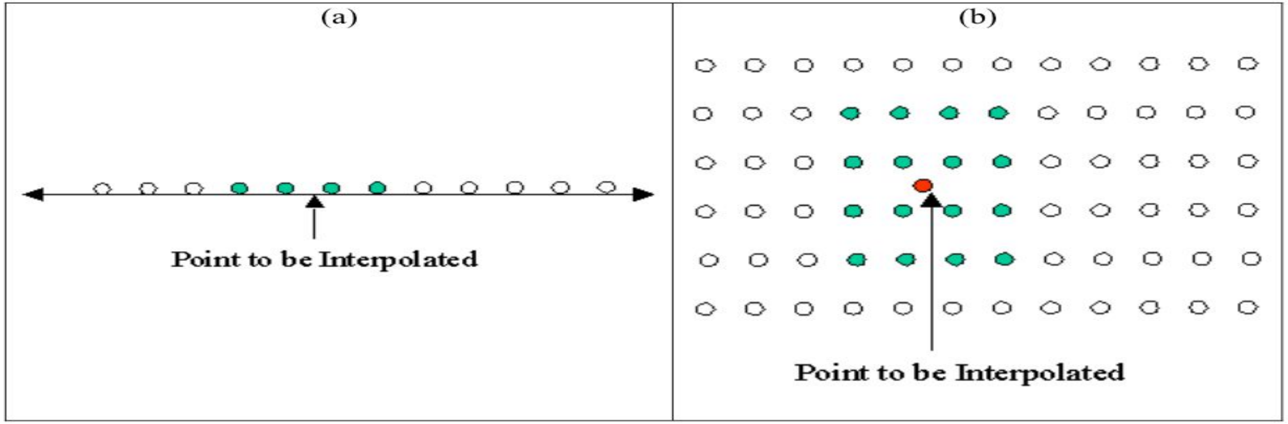
$AI_4+BI_3+CI_2+DI_1$

$A+B+C+D = 1$

Interpolación de intensidades

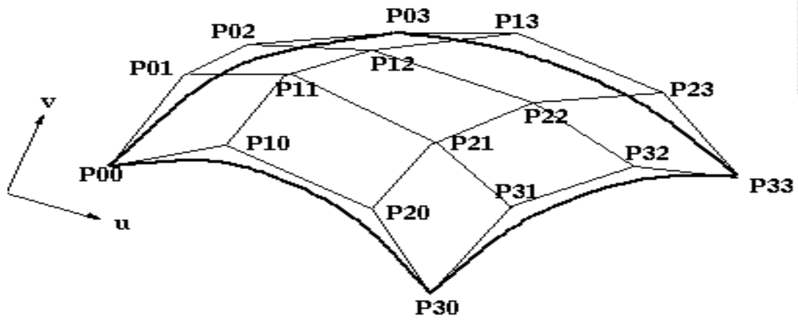
- Interpolación de Intensidades
  - Interpolación bicúbica

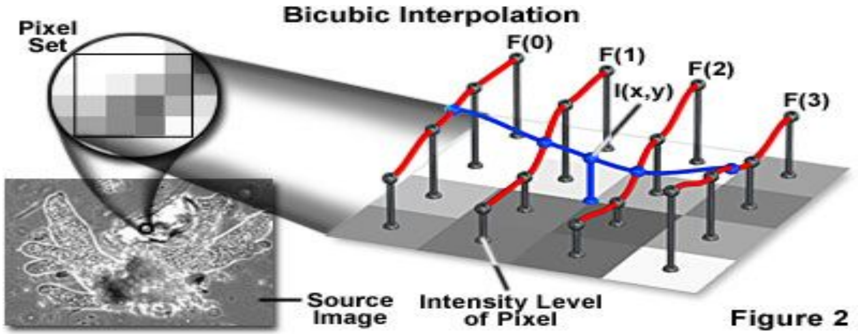
Requiere 16 puntos, vecindad de 4x4



Interpolación de intensidades

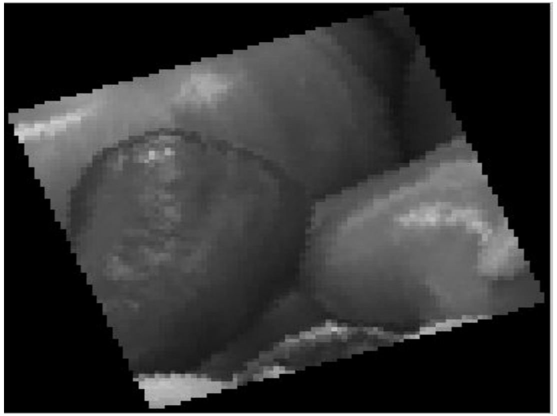
- Interpolación de Intensidades
  - Interpolación bicúbica



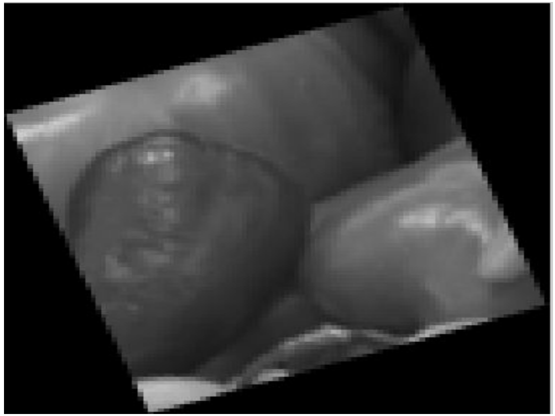


Interpolación de intensidades

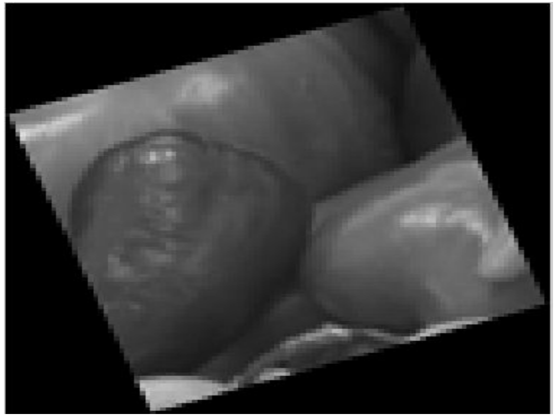
- Interpolación de Intensidades
  - Ejemplo



Vecino más cercano



Bilineal



Bicúbica

Transformaciones Geométricas

- Clasificación de Transformaciones
  - Transformación Euclidiana (Isometría)
    - Rotación + Traslación

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} cos(\theta) & -sin(\theta) & t_x \\ sin(\theta) & cos(\theta) & t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

    - Transformación de cuerpos rígidos

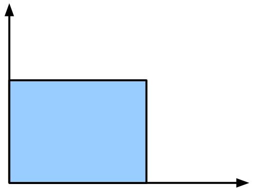
Transformaciones Geométricas

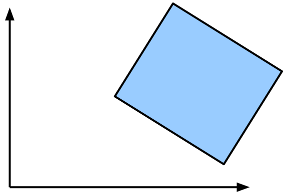
- Clasificación de Transformaciones
  - Transformación Euclidiana (Isometría)

Invariante a:

- Distancia entre puntos (longitud).
- Ángulos.
- Área

Grado de libertad: 3





Transformaciones Geométricas

- Clasificación de Transformaciones
  - Transformación de similitud (similarity transformation)
  - Escalamiento + Rotación + Traslación

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} s \cdot cos(\theta) & -s \cdot sin(\theta) & t_x \\ s \cdot sin(\theta) & s \cdot cos(\theta) & t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

Escalamiento isométrico

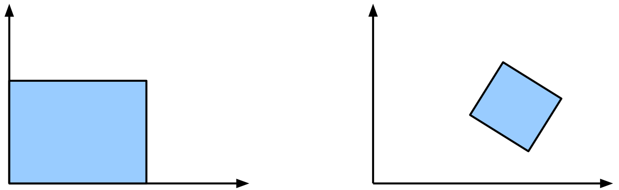
Transformaciones Geométricas

- Clasificación de Transformaciones
  - Transformación de similitud (similarity transformation)

Invariante a:

- Ángulos.
- Líneas paralelas.
- Razón entre longitudes.
- Razón de áreas de objetos.

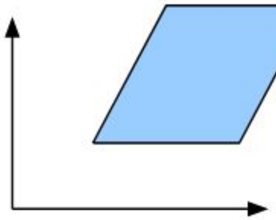
Grados de Libertad: 4



Transformaciones Geométricas

- Clasificación de Transformaciones
  - Transformación Afín (affine transformation)
  - Escalamiento no isométrico + Rotación + Traslación

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

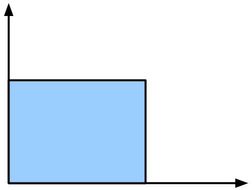


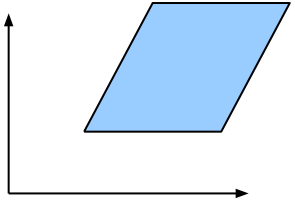
Transformaciones Geométricas

- Clasificación de Transformaciones
  - Transformación Afín (affine transformation)

Invariante a:
- Colinealidad de puntos.
- Líneas paralelas.
- Razón entre longitudes de segmentos de líneas paralelas.
- Razón de áreas de objetos.

Grados de Libertad: 6

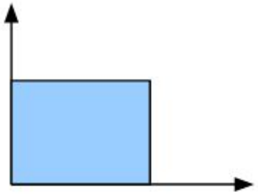


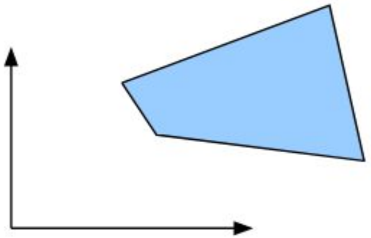


Transformaciones Geométricas

- Clasificación de Transformaciones
  - Transformaciones de Proyección (homografía)

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$



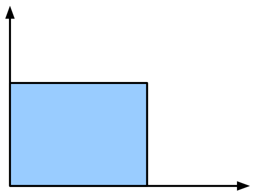
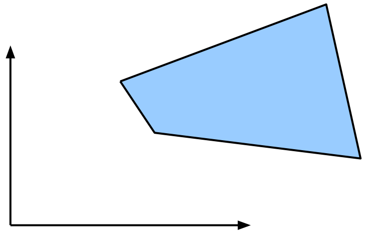


Transformaciones Geométricas

- Clasificación de Transformaciones
  - Transformaciones de Proyección (homografía)

Invariante a:

- Colinealidad de puntos (líneas rectas).
- **Grados de Libertad: 8**

Transformaciones Geométricas

- Clasificación de Transformaciones
  - Transformaciones de Proyección (homografía)

Transformaciones Geométricas

- Clasificación de Transformaciones
  - Transformaciones de Proyección (homografía)

Eliminando distorsiones de proyección

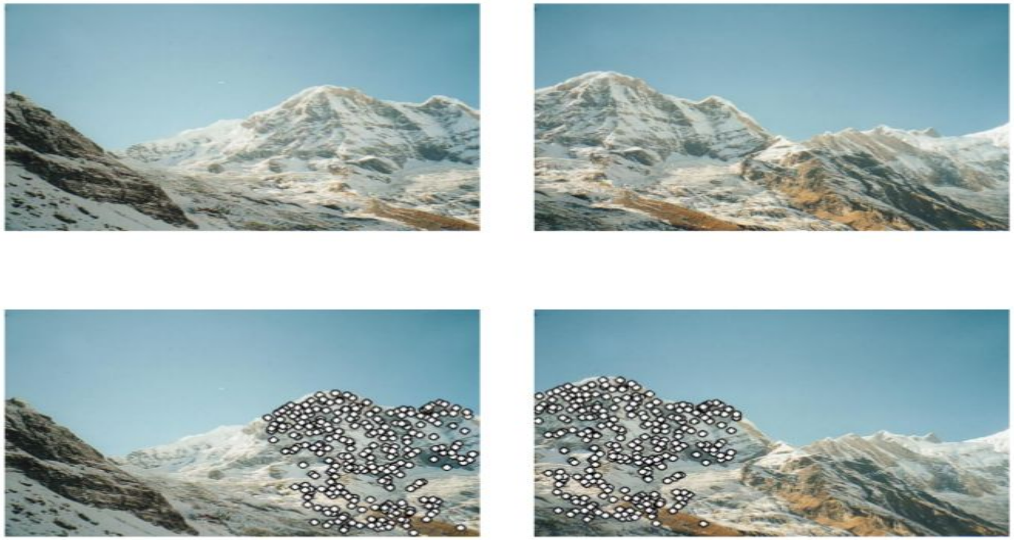
Transformaciones Geométricas

- Clasificación de Transformaciones
  - Transformaciones de Proyección (homografía)

Eliminando distorsiones de proyección

Transformaciones Geométricas

- Clasificación de Transformaciones
  - Transformaciones de Proyección (homografía)



Stitching

Transformaciones Geométricas

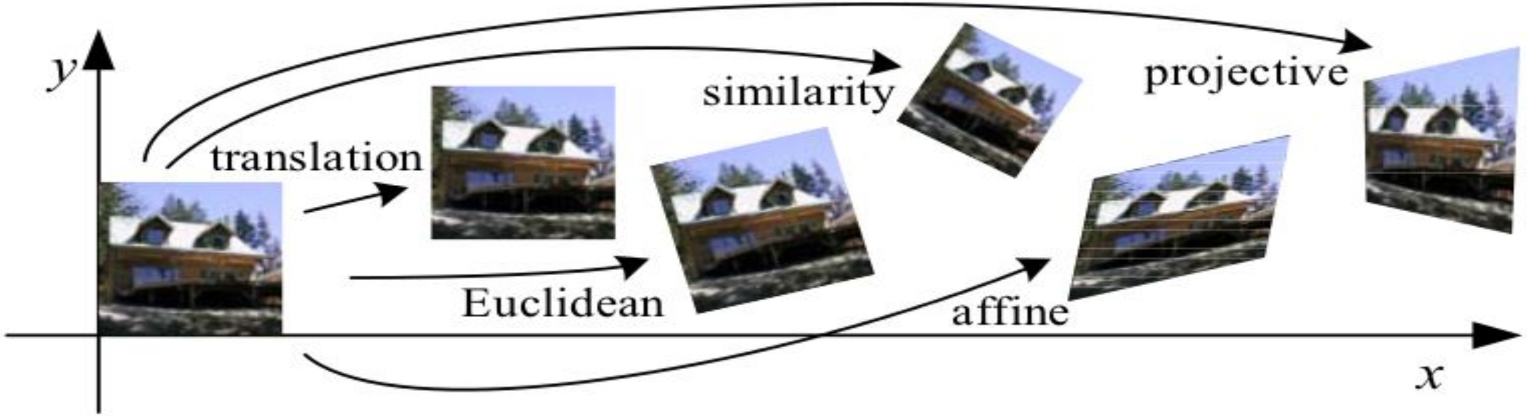
- Clasificación de Transformaciones
  - Transformaciones de Proyección (homografía)

Stitching

[Brown, Lowe] Recognising Panoramas ICCV 2003

Transformaciones Geométricas

- Clasificación de Transformaciones



Transformaciones Geométricas

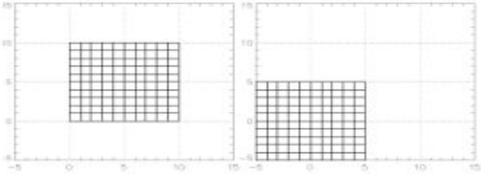
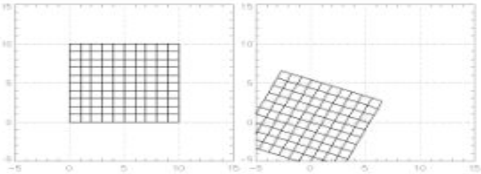
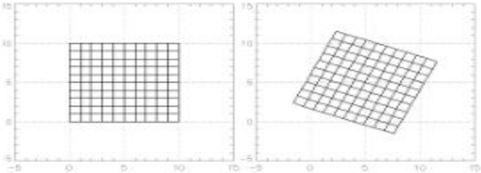
Homogeneous Coordinates

- Notation useful for converting scaling, translation, rotating into point-matrix multiplication
- To convert ordinary coordinates into homogeneous coordinates

$$\boldsymbol{x} = \begin{pmatrix} x \\ y \end{pmatrix} \quad \text{converts to} \quad \hat{\boldsymbol{x}} = \begin{pmatrix} \hat{x} \\ \hat{y} \\ h \end{pmatrix} = \begin{pmatrix} h\,x \\ h\,y \\ h \end{pmatrix}$$

Transformaciones Geométricas

Combined Transform Operations

Operation	Expression	Result
Translate to Origin	$\mathbf{T}_1 = \begin{bmatrix} 1.00 & 0.00 & -5.00 \\ 0.00 & 1.00 & -5.00 \\ 0.00 & 0.00 & 1.00 \end{bmatrix}$	
Rotate by 23 degrees	$\mathbf{T}_2 = \begin{bmatrix} 0.92 & 0.39 & 0.00 \\ -0.39 & 0.92 & 0.00 \\ 0.00 & 0.00 & 1.00 \end{bmatrix}$	
Translate to original location	$\mathbf{T}_3 = \begin{bmatrix} 1.00 & 0.00 & 5.00 \\ 0.00 & 1.00 & 5.00 \\ 0.00 & 0.00 & 1.00 \end{bmatrix}$	

Transformaciones Geométricas

The transformation matrix of a sequence of affine transformations, say $\mathbf{T}_1$ then $\mathbf{T}_2$ then $\mathbf{T}_3$ is

$$\mathbf{T} = \mathbf{T}_3\mathbf{T}_2\mathbf{T}_3$$

The composite transformation for the example above is

$$\mathbf{T} = \mathbf{T}_3\mathbf{T}_2\mathbf{T}_1 = \begin{bmatrix} 0.92 & 0.39 & -1.56 \\ -0.39 & 0.92 & 2.35 \\ 0.00 & 0.00 & 1.00 \end{bmatrix}$$

Any combination of affine transformations formed in this way is an affine transformation.

The inverse transform is

$$\mathbf{T}^{-1} = \mathbf{T}_1^{-1}\mathbf{T}_2^{-1}\mathbf{T}_3^{-1}$$

If we find the transform in one direction, we can invert it to go the other way.

Transformaciones Geométricas

Suppose that you want the composite representation for translation, scaling and rotation (in that order).

$$
\begin{aligned}
H = RST &= \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_0 & 0 & 0 \\ 0 & s_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & x_0 \\ 0 & 1 & x_1 \\ 0 & 0 & 1 \end{bmatrix} \\
&= \begin{bmatrix} s_0\cos\theta & s_1\sin\theta & s_0x_0\cos\theta + s_1x_1\sin\theta \\ -s_0\sin\theta & s_1\cos\theta & s_1x_1\cos\theta - s_0x_0\sin\theta \end{bmatrix}
\end{aligned}
$$

Given the matrix H one can solve for the five parameters.

Transformaciones Geométricas

How to Find the Transformation

Suppose that you are given a pair of images to align. You want to try an affine transform to register one to the coordinate system of the other. How do you find the transform parameters?

Image A

Image B

Transformaciones Geométricas

Point Matching

Find a number of points $\{\mathbf{p}_0, \mathbf{p}_1, \ldots, \mathbf{p}_{n-1}\}$ in image A that match points $\{\mathbf{q}_0, \mathbf{q}_1, \ldots, \mathbf{q}_{n-1}\}$ in image B. Use the homogeneous coordinate representation of each point as a column in matrices $\mathbf{P}$ and $\mathbf{Q}$:

$$\mathbf{P} = \begin{bmatrix} x_0 & x_1 & \ldots & x_{n-1} \\ y_0 & y_1 & \ldots & y_{n-1} \\ 1 & 1 & \ldots & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{p}_0 & \mathbf{p}_1 & \ldots & \mathbf{p}_{n-1} \end{bmatrix}$$

$$\mathbf{Q} = \begin{bmatrix} u_0 & u_1 & \ldots & u_{n-1} \\ v_0 & v_1 & \ldots & v_{n-1} \\ 1 & 1 & \ldots & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{q}_0 & \mathbf{q}_1 & \ldots & \mathbf{q}_{n-1} \end{bmatrix}$$

Then

$$\mathbf{q} = \mathbf{H}\mathbf{p} \quad \text{becomes} \quad \mathbf{Q} = \mathbf{H}\mathbf{P}$$

The solution for $\mathbf{H}$ that provides the minimum mean-squared error is

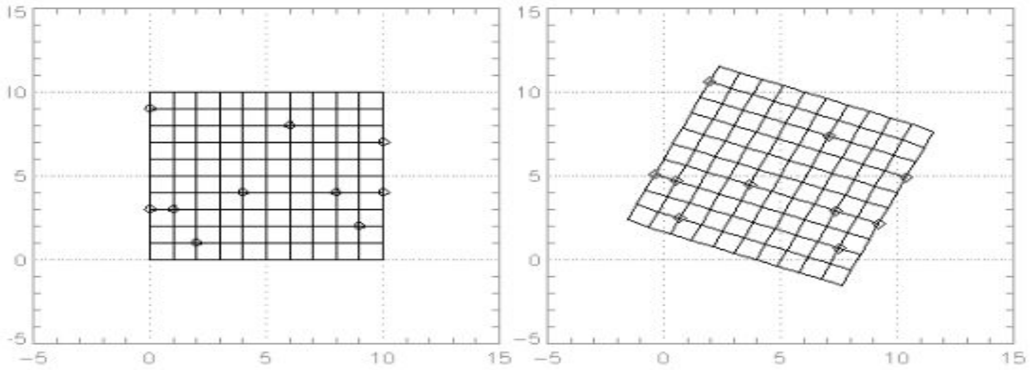
$$\mathbf{H} = \mathbf{Q}\mathbf{P}^T(\mathbf{P}\mathbf{P}^T)^{-1} = \mathbf{Q}\mathbf{P}^\dagger$$

where $\mathbf{P}^\dagger = \mathbf{P}^T(\mathbf{P}\mathbf{P}^T)^{-1}$ is the (right) *pseudo-inverse* of $\mathbf{P}$.

Transformaciones Geométricas

Point Matching

The transformation used in the previous example can be found from a few matching points chosen randomly in each image.



Many image processing tools, such as ENVI, have tools to enable point-and-click selection of matching points.