Transformation linéaires

GIF-4105/7105 Photographie Algorithmique, Hiver 2015

Jean-François Lalonde

Merci à D. Hoiem, A. Efros et S. Seitz

Cette semaine

- Aujourd'hui:
  - Transformations linéaires globales
  - Calculer la transformation à partir d'images
  - Appliquer une transformation à une image
  - Morphage

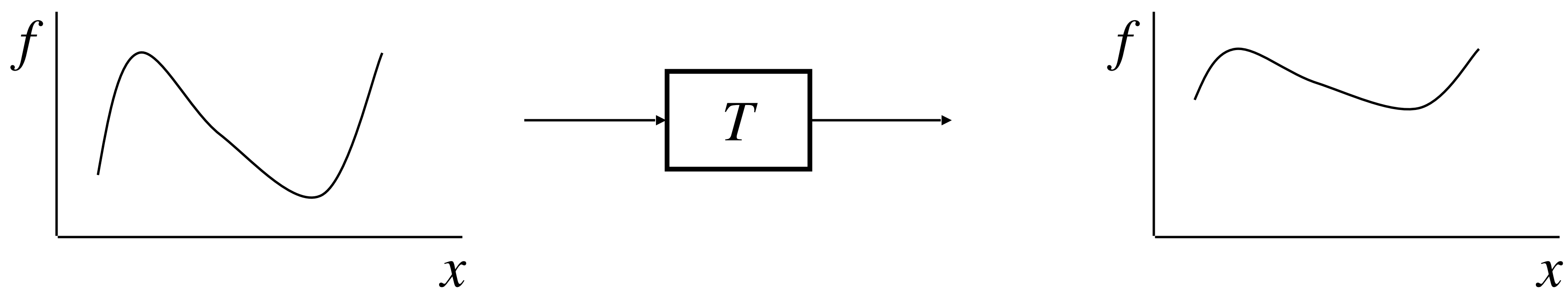
Avant tout!

- TP3 disponible sur le site web du cours
- À la pause: photos

Transformations d’image

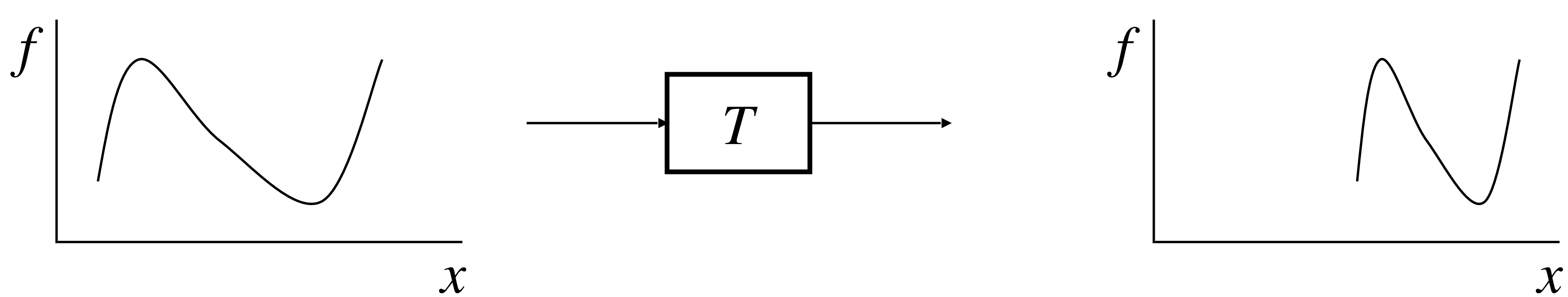
- filtrage: modifier l'image du signal

$$g(x) = T(f(x))$$



- transformations: modifier le domaine du signal

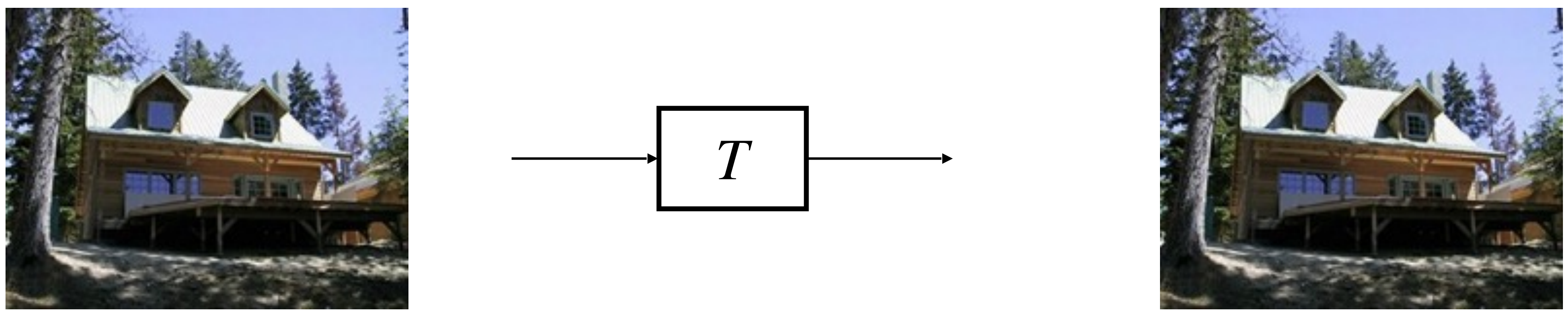
$$g(x) = f(T(x))$$



Transformations d’image

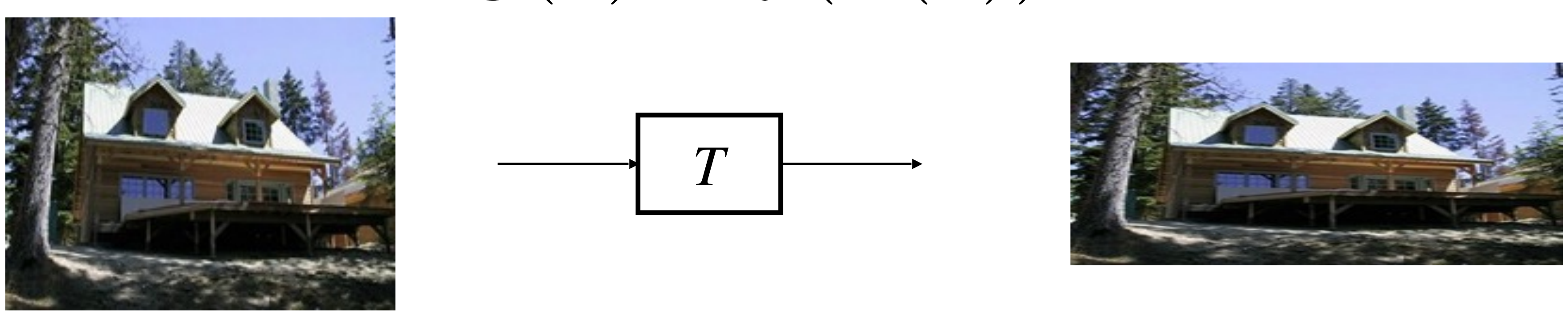
- filtrage: modifier l’image du signal

$$g(x) = T(f(x))$$



- transformations: modifier le domaine du signal

$$g(x) = f(T(x))$$



Transformations globales (paramétriques)

- Transformation $\boldsymbol{T}$ modifie les coordonnées:

$\mathbf{p} = (x, y)$

$$\longrightarrow \boxed{T} \longrightarrow$$

$\mathbf{p}' = (x', y')$

$$\mathbf{p}' = T(\mathbf{p})$$

- Qu'est-ce que "globale" veut dire?
  - La même chose pour chaque point
  - Peut être représentée par un faible nombre de paramètres (paramétrique)
- Pour les transformations linéaires, on peut représenter la transformation par une matrice:

$$\mathbf{p}' = \mathbf{M}\mathbf{p} \qquad \begin{bmatrix} x' \\ y' \end{bmatrix} = \mathbf{M} \begin{bmatrix} x \\ y \end{bmatrix}$$

Transformations globales (paramétriques)

- Exemples:

translation

rotation

aspect

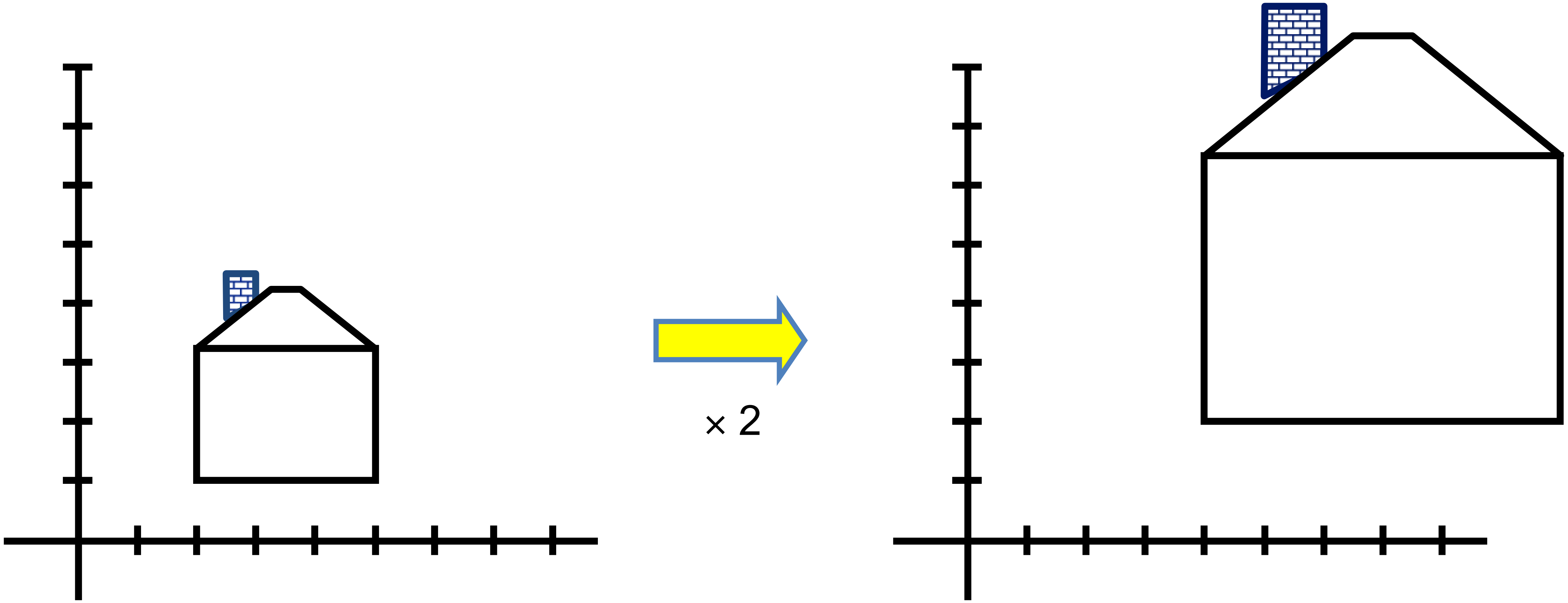
affine

perspective

cylindrique

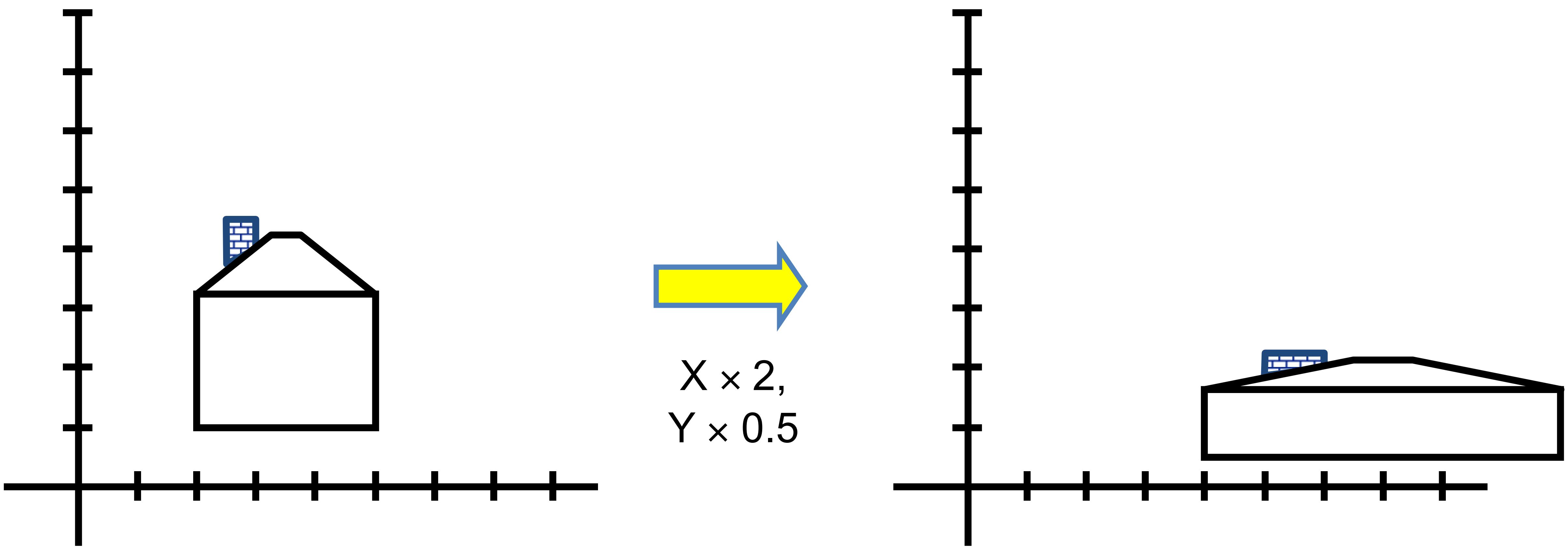
Mise à l'échelle

- Multiplier chaque coordonnée par un scalaire
- Uniforme: le même scalaire pour chaque coordonnées (ici: x et y)



Mise à l’échelle

- Non-uniforme: différent scalaire par coordonnée



Mise à l'échelle

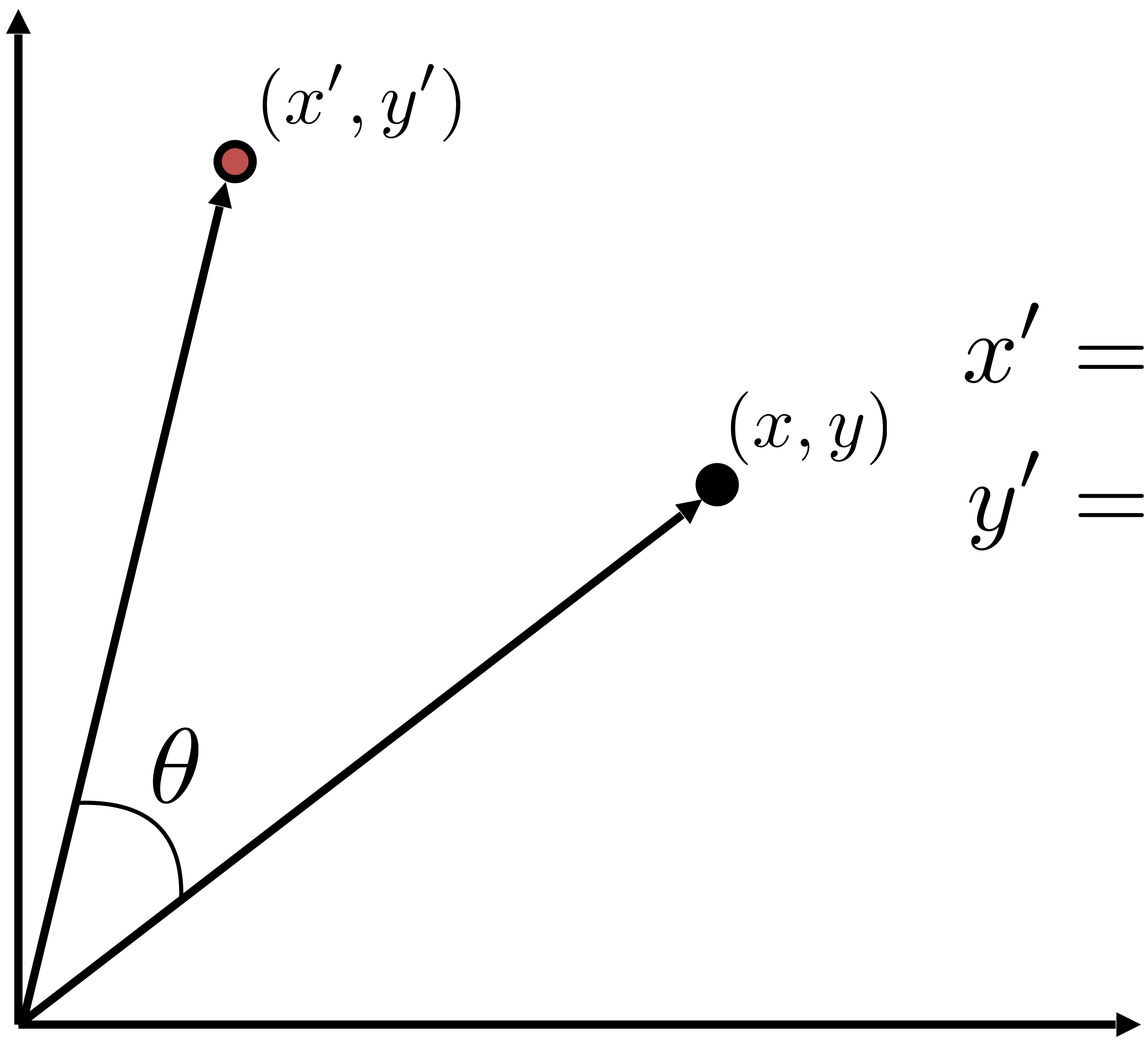
- Opération:

$$x' = ax$$
$$y' = ay$$

- Matrice:

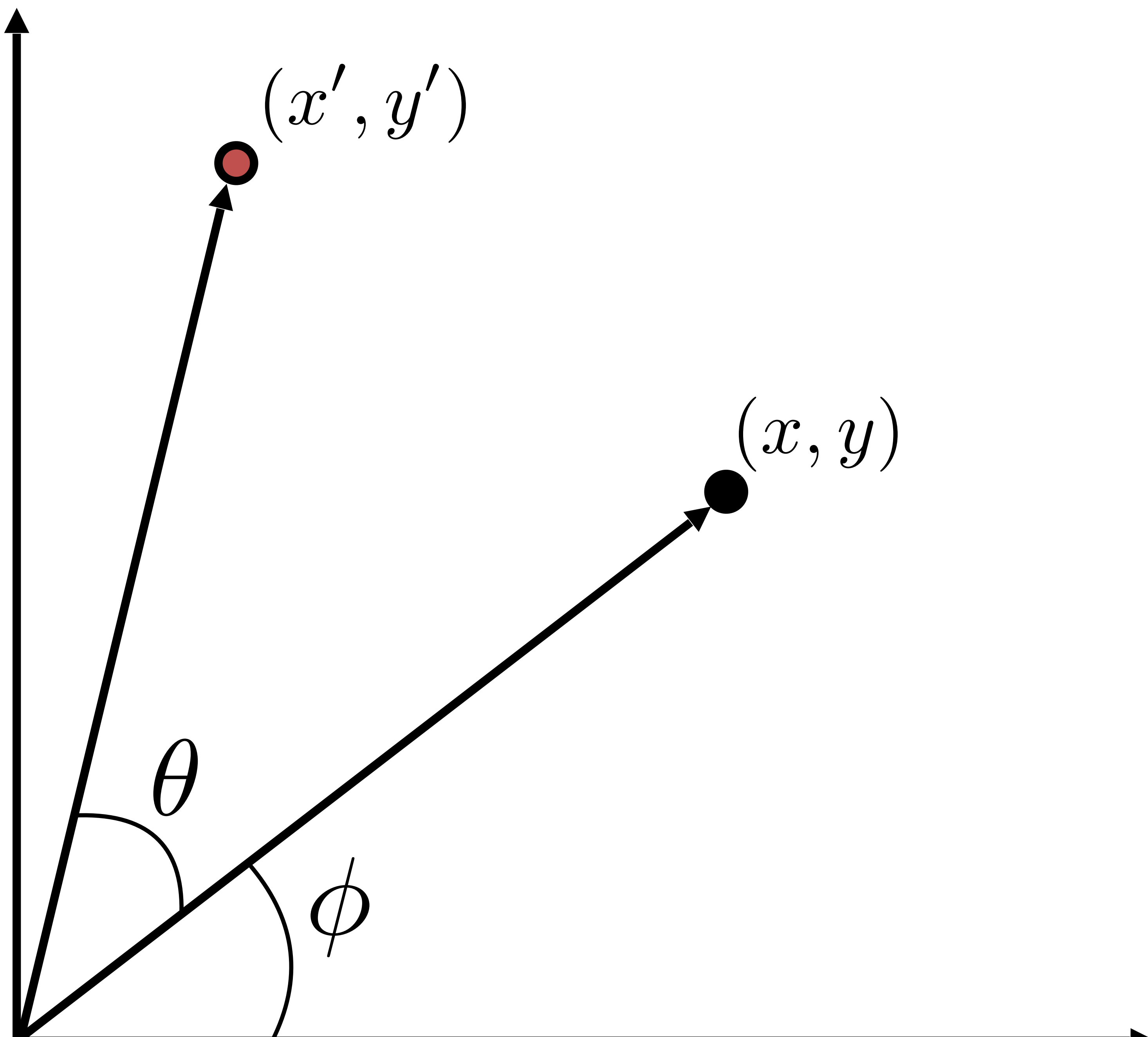
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \underbrace{\begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix}}_{\text{matrice } \mathbf{S}} \begin{bmatrix} x \\ y \end{bmatrix}$$

Rotation 2D



$$x' = x\cos\theta - y\sin\theta$$
$$y' = x\sin\theta + y\cos\theta$$

Rotation 2D



Coordonnées polaires

x = r cos (ϕ)

y = r sin (ϕ)

x’ = r cos (ϕ + θ)

y’ = r sin (ϕ + θ)

Identité trigonométrique

x’ = r cos(ϕ) cos(θ) – r sin(ϕ) sin(θ)

y’ = r sin(ϕ) cos(θ) + r cos(ϕ) sin(θ)

Substitution

x’ = x **cos**(θ) - y **sin**(θ)

y’ = x **sin**(θ) + y **cos**(θ)

Rotation 2D

- Forme matricielle:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \underbrace{\begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}}_{\mathbf{R}} \begin{bmatrix} x \\ y \end{bmatrix}$$

- Même si sin(θ) et cos(θ) sont des fonctions non-linéaires en θ,
  - x' et y' sont des combinaisons linéaires de x et y
- Quelle est la transformation inverse?
  - Rotation par –θ
  - Pour les matrices de rotation:

$$\mathbf{R}^{-1} = \mathbf{R}^T$$

Matrices 2x2

- Quelles transformations peuvent être représentées par des matrices 2x2?

Identité?

$$x' = x$$
$$y' = y$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Facteur d'échelle autour de (0,0)?

$$x' = s_x x$$
$$y' = s_y y$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} s_x & 0 \\ 0 & x_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Matrices 2x2

- Quelles transformations peuvent être représentées par des matrices 2x2?

Réflexion en x?

$$x' = -x$$
$$y' = y$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Réflexion par rapport à l'origine?

$$x' = -x$$
$$y' = -y$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Matrices 2x2

- Quelles transformations peuvent être représentées par des matrices 2x2?

Rotation?

$$x' = x\cos\theta - y\sin\theta$$
$$y' = x\sin\theta + y\cos\theta$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Étirement (shear)?

$$x' = x + k_x y$$
$$y' = y + k_y x$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & k_x \\ k_y & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Matrices 2x2

- Quelles transformations peuvent être représentées par des matrices 2x2?

Translation?

$$x' = x + t_x$$
$$y' = y + t_y$$

NON!

Seulement les fonctions linéaires en x et y peuvent être représentées par des matrices 2x2

Transformations linéaires

- Toutes les transformations linéaires sont des combinaisons de:
    - échelle, rotation, étirement, réflexion $$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$
- Propriétés
    - Origine ne change pas
    - Sont préservés: $$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} e & f \\ g & h \end{bmatrix} \begin{bmatrix} i & j \\ k & l \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$
        - Lignes, lignes parallèles, ratios
    - Composition est aussi une transformation linéaire

Translations?

- Comment pouvons-nous représenter les translations sous forme matricielle?

$$x' = x + t_x$$
$$y' = y + t_y$$

Coordonnées homogènes

- Représente des coordonnées 2-D avec un vecteur à 3 éléments

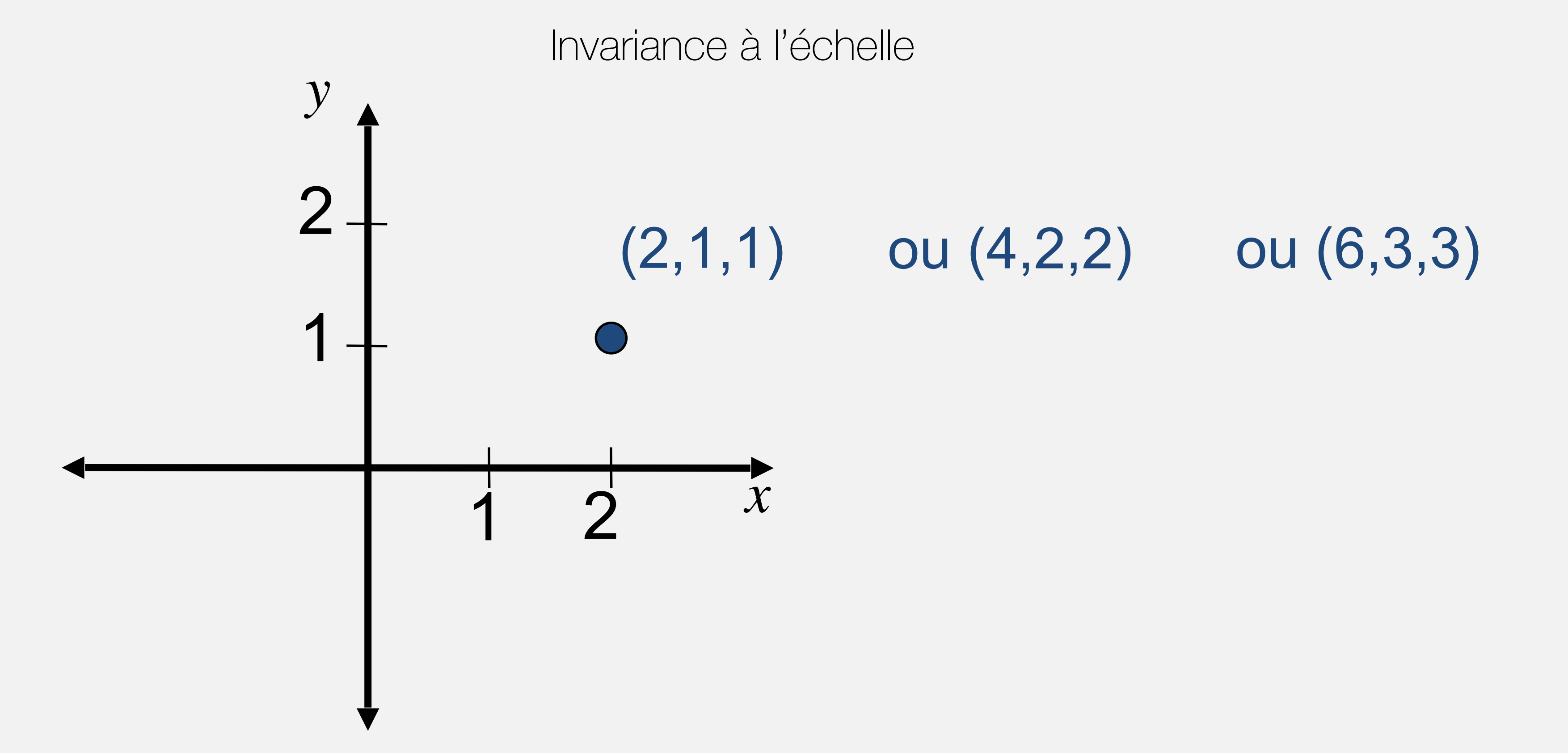
$$\begin{bmatrix} x \\ y \end{bmatrix} \xrightarrow{\text{Coordonnées homogènes}} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} x \\ y \\ w \end{bmatrix} \xrightarrow{\text{Point 2D}} \begin{bmatrix} x/w \\ y/w \end{bmatrix}$$

Coordonnées homogènes

- Propriétés:
  - Invariance au facteur d'échelle
  - (x, y, 0) représente un point à l'infini
  - (0, 0, 0) n'est pas permis

$$\begin{bmatrix} x \\ y \\ w \end{bmatrix} = k \begin{bmatrix} x \\ y \\ w \end{bmatrix}$$



Translations?

- Comment pouvons-nous représenter les translations sous forme matricielle?

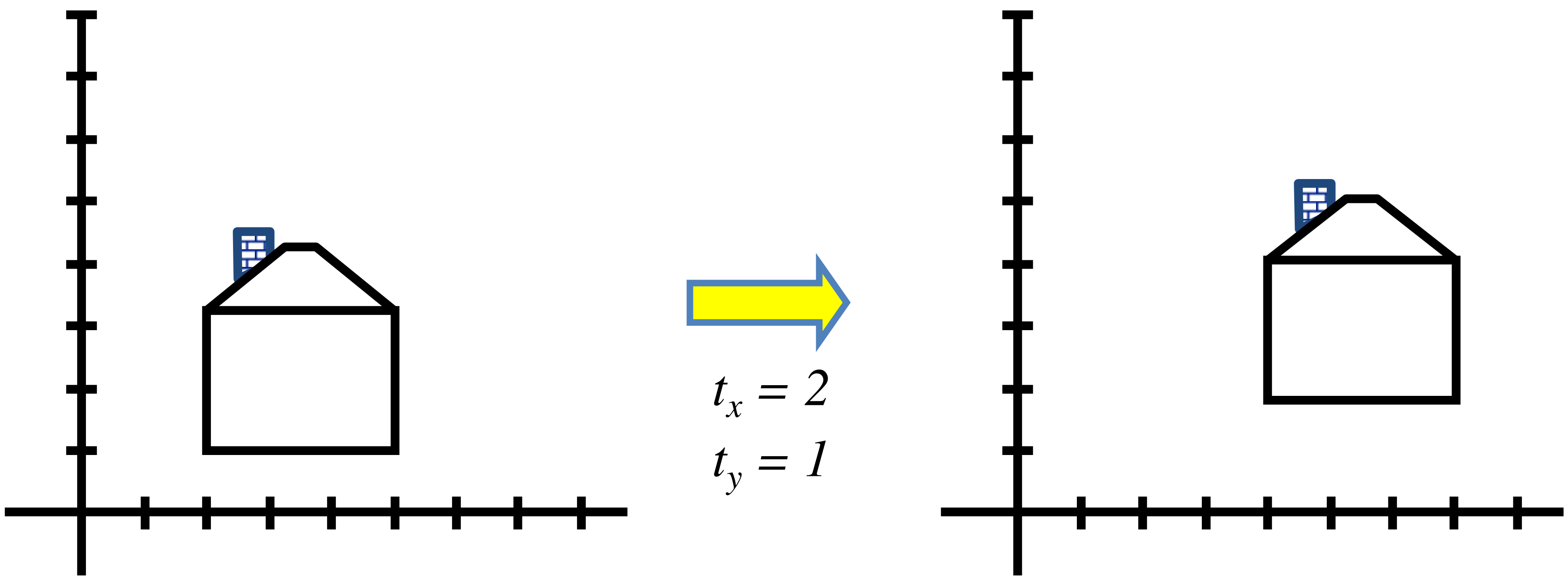
$$x' = x + t_x$$
$$y' = y + t_y$$

- En utilisant une troisième colonne!

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Exemple de translation

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$



Transformations 2D en matrices 3x3

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Translation

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Échelle

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Rotation

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & k_x & 0 \\ k_y & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Étirement

Composition

- Les transformations peuvent être composées en multipliant les matrices

$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_x & 0 & 0 \\ 0 & x_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ w \end{bmatrix}$$

Démonstration Matlab

Transformations affines

- Transformées affines sont des combinaisons de:
  - Transformées linéaires; et
  - Translations

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

- Propriétés
  - L'origine n'est pas nécessairement préservée
  - Sont préservées: les lignes, lignes parallèles, ratios
  - Composition est aussi une transformée affine

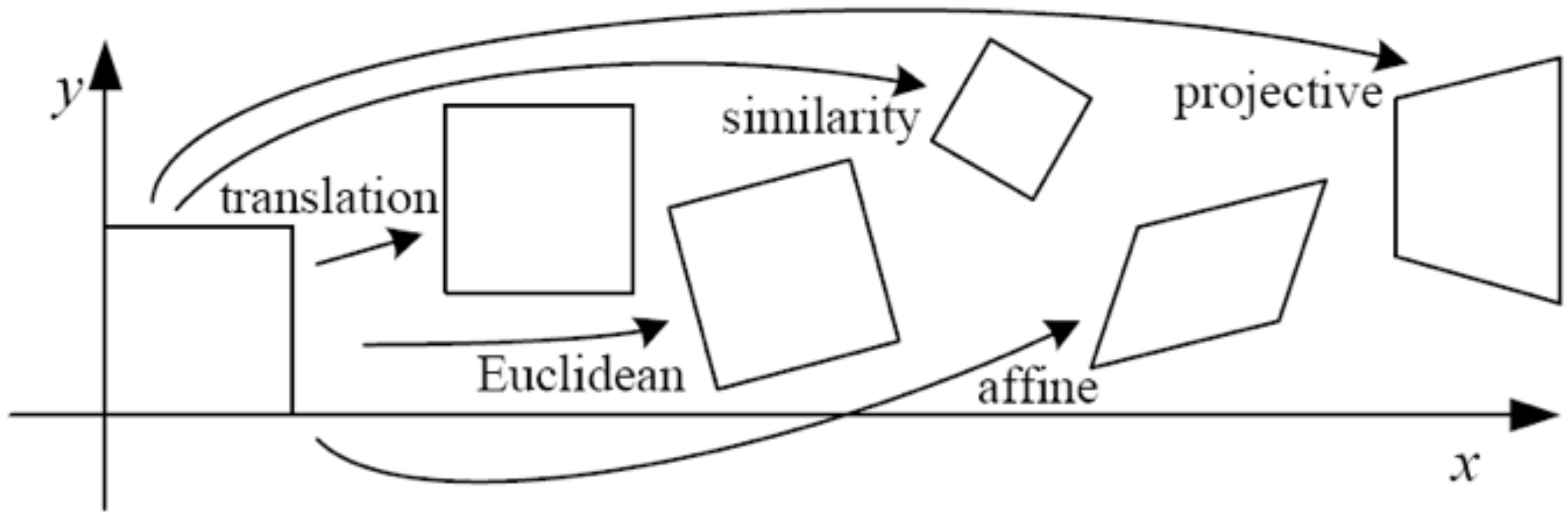
Transformations projectives

- Transformées affines sont des combinaisons de:
  - Transformées affines; et
  - Projections

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

- Propriétés
  - L'origine n'est pas nécessairement préservée
  - Sont préservées: les lignes, ~~lignes parallèles, ratios~~
  - Composition est aussi une transformée affine
  - Définies jusqu'à un facteur d'échelle (8 DDL)

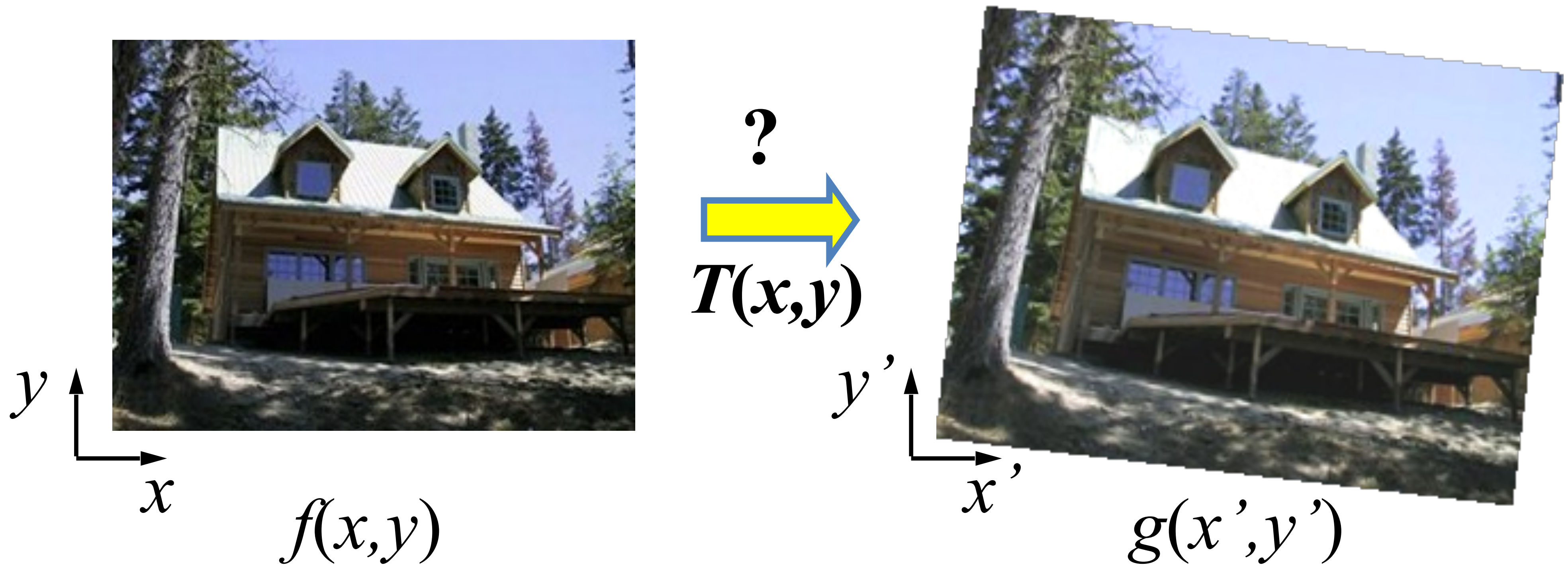
Transformations en 2D



Name	Matrix	# D.O.F.	Preserves:	Icon
translation	$\left[\, \boldsymbol{I} \mid \boldsymbol{t} \,\right]_{2\times 3}$			
rigid (Euclidean)	$\left[\, \boldsymbol{R} \mid \boldsymbol{t} \,\right]_{2\times 3}$			
similarity	$\left[\, s\boldsymbol{R} \mid \boldsymbol{t} \,\right]_{2\times 3}$			
affine	$\left[\, \boldsymbol{A} \,\right]_{2\times 3}$			
projective	$\left[\, \tilde{\boldsymbol{H}} \,\right]_{3\times 3}$			

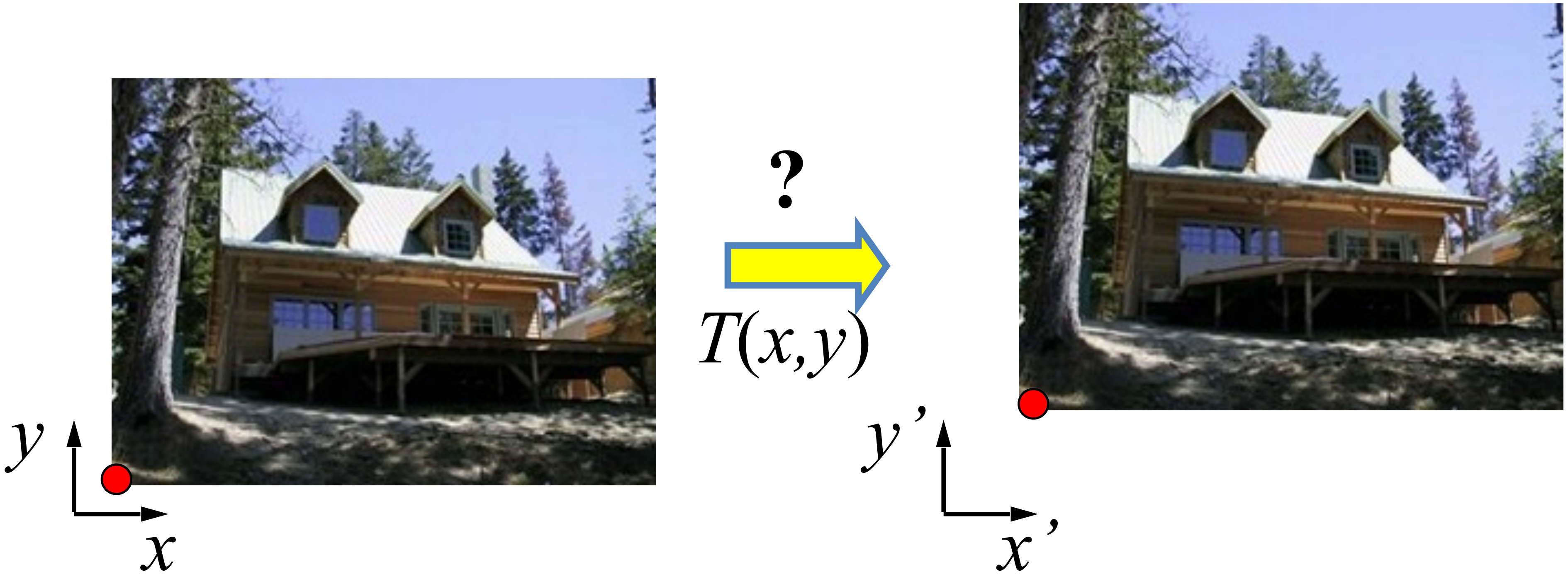
Composition et inverse font aussi parties du groupe

Estimer les transformations



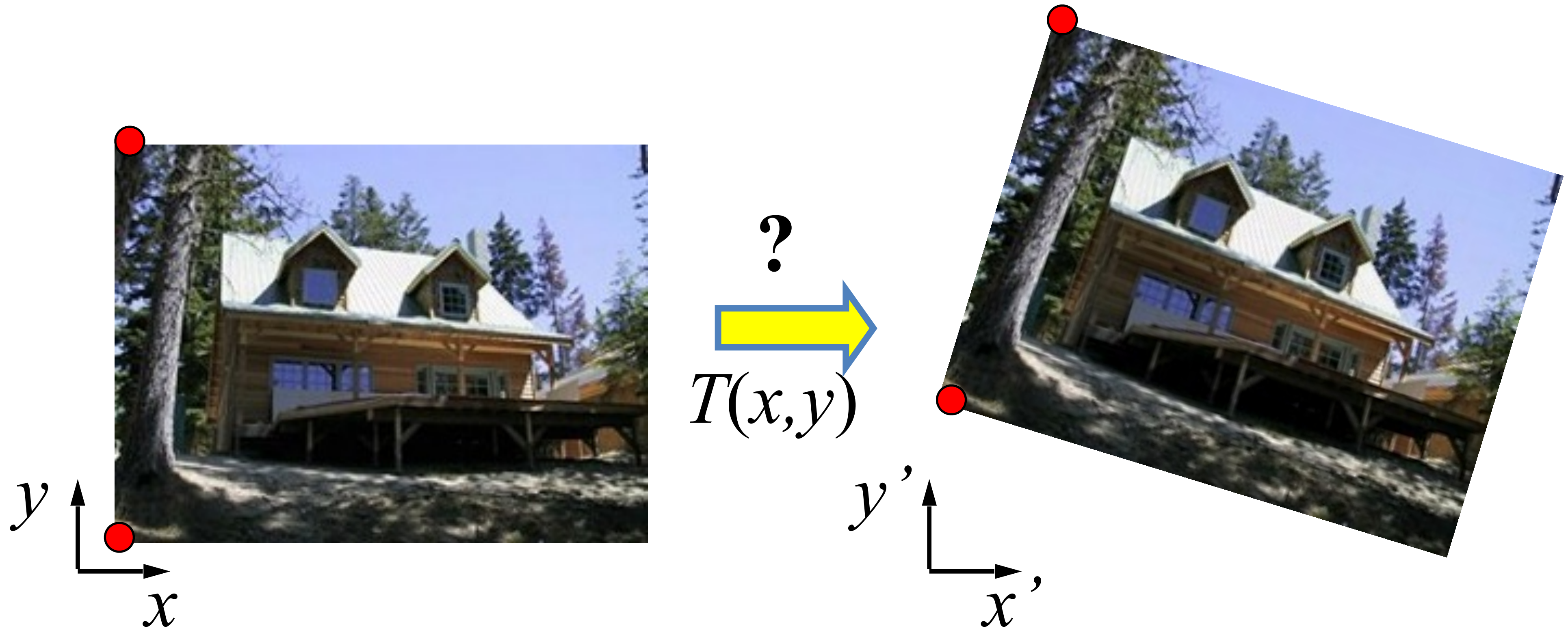
- Admettons que nous connaissons deux images (f et g). Comment faire pour estimer leur transformation?
- Demandons à un utilisateur de nous donner des correspondences
  - Combien en avons-nous besoin?

Translation



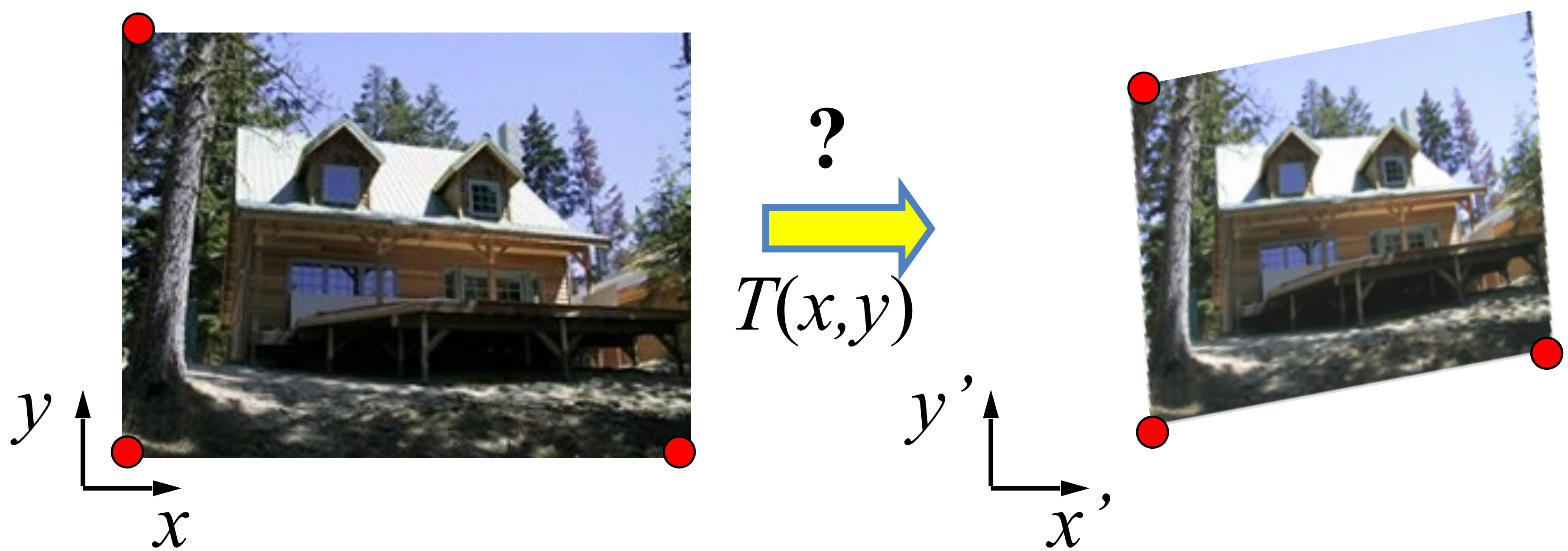
- Combien de degrés de liberté (DDL)?
- Combien de correspondences?

Euclidienne



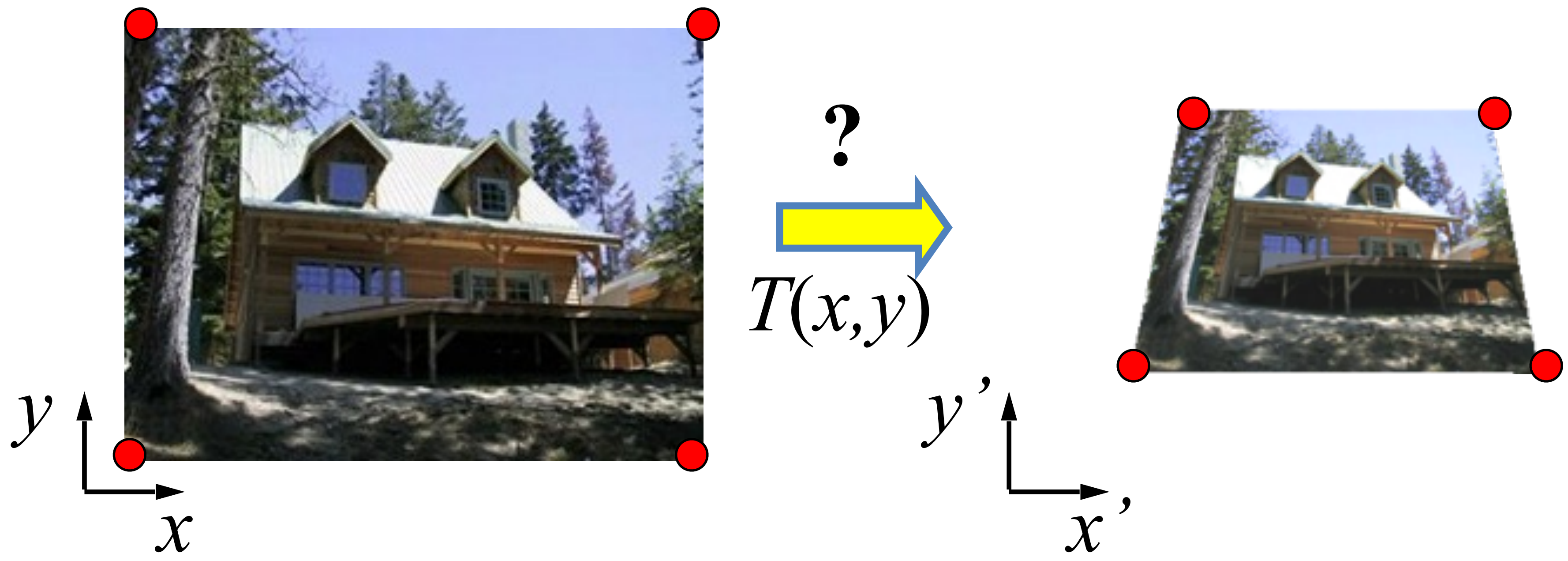
- Combien de degrés de liberté (DDL)?
- Combien de correspondences?

Affine



- Combien de degrés de liberté (DDL)?
- Combien de correspondences?

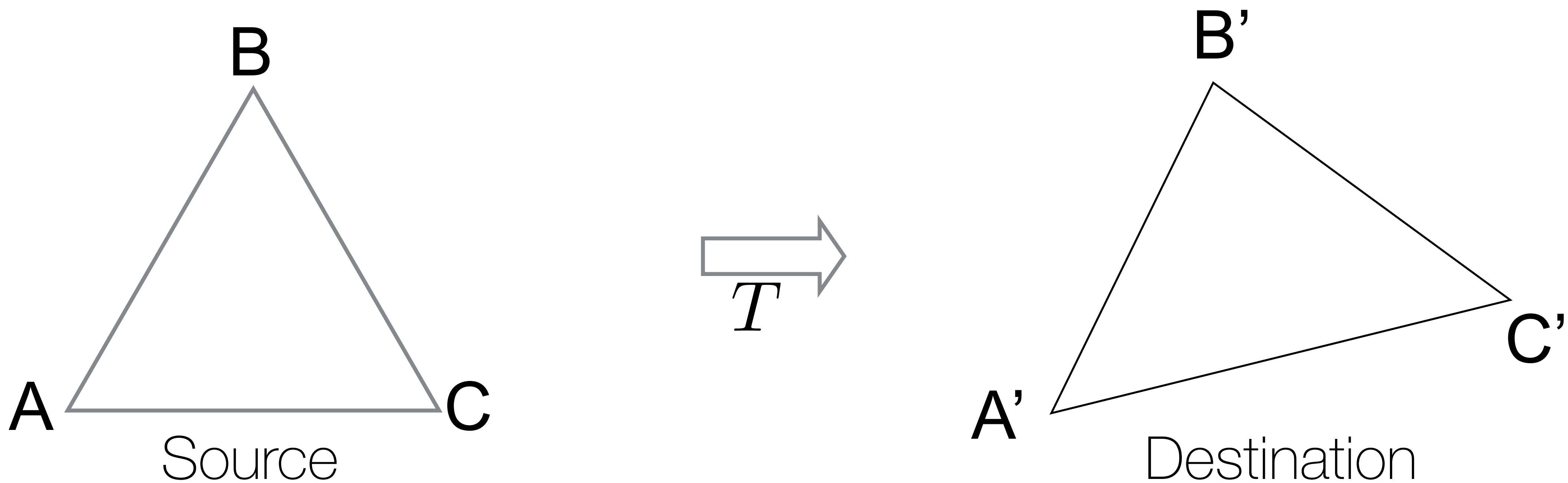
Projective



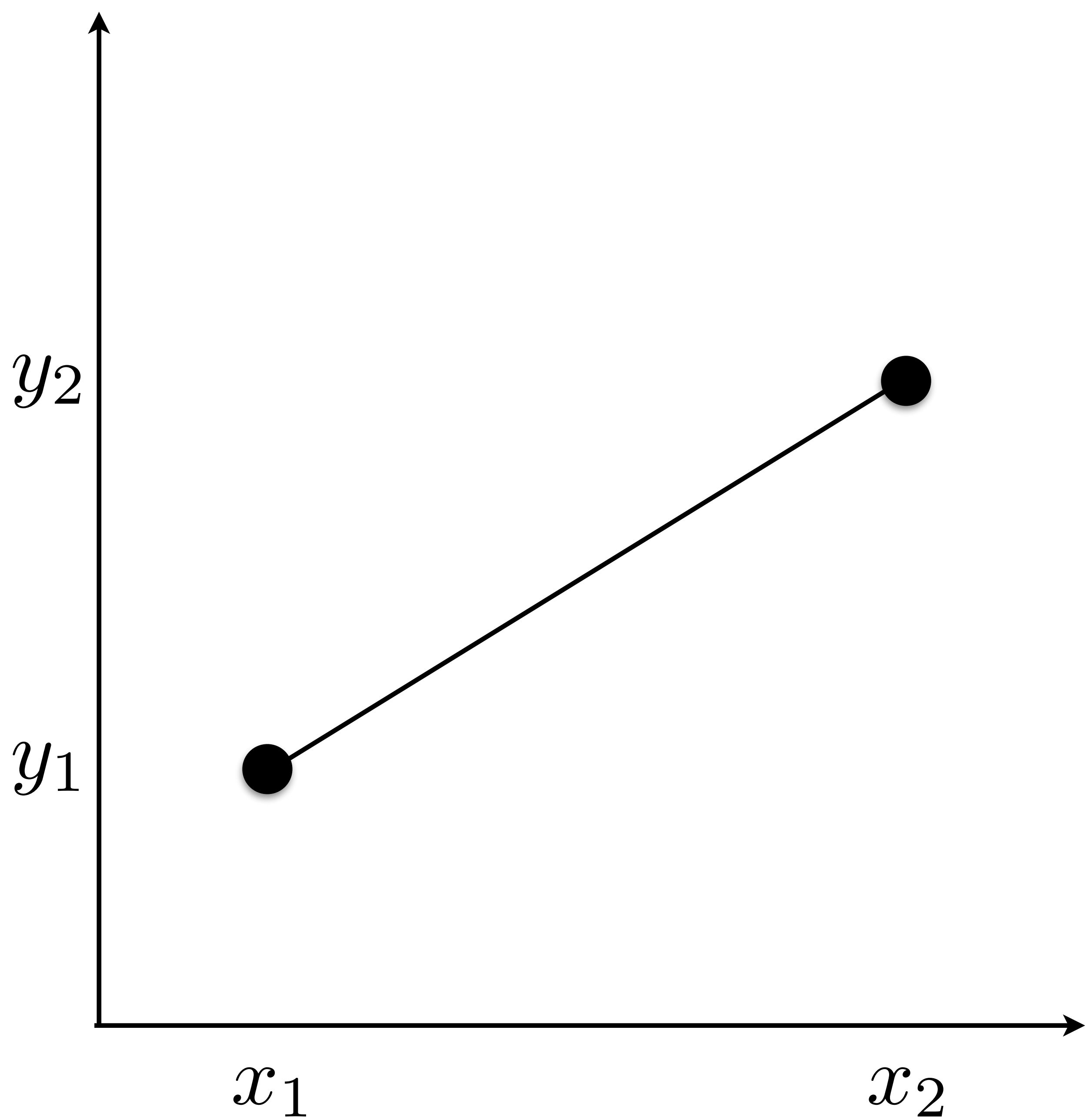
- Combien de degrés de liberté (DDL)?
- Combien de correspondences?

Questions

- Supposons que nous avons deux triangles:
  - ABC et A’B’C’
- Quelle est la transformation qui passe de ABC vers A’B’C’?
- Comment pouvons-nous estimer ses paramètres?

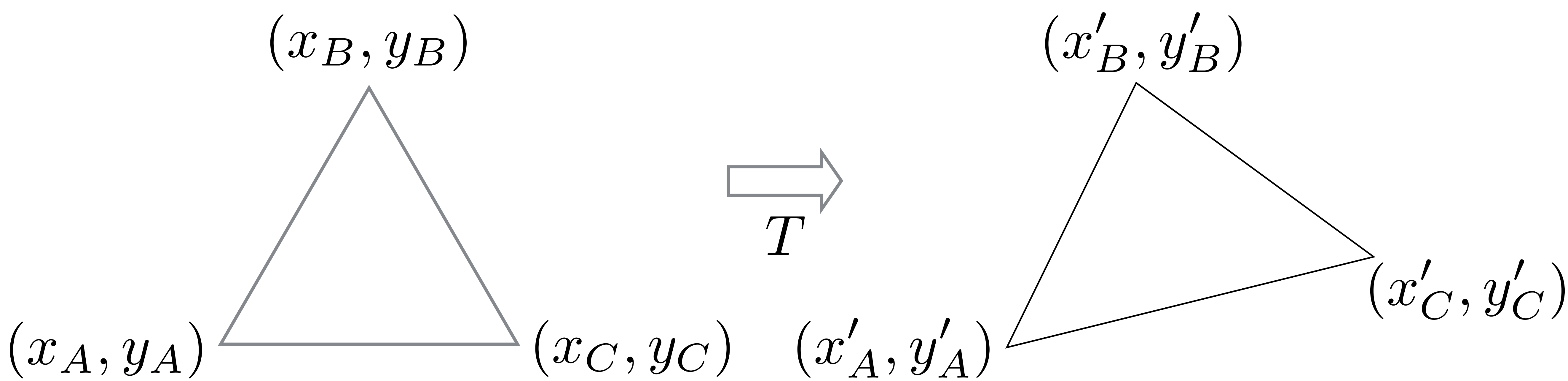


Estimation de paramètres



$$y = ax + b$$

Estimation de paramètres

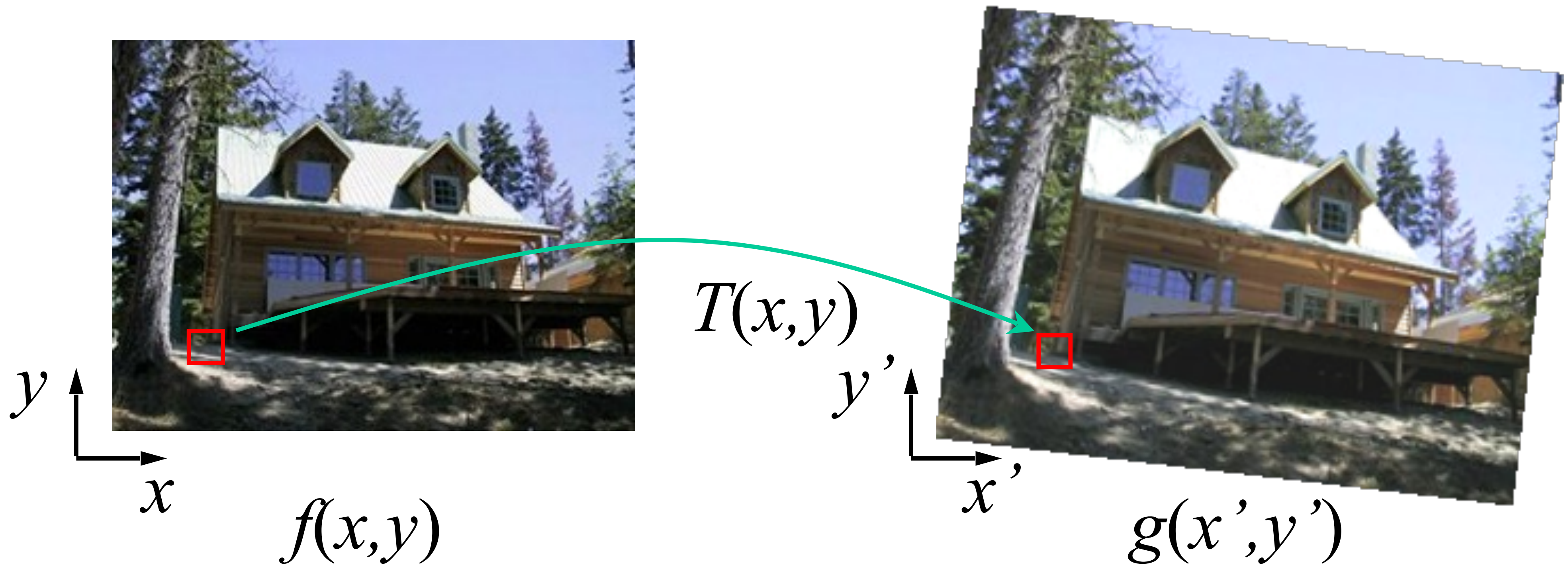


Source

Destination

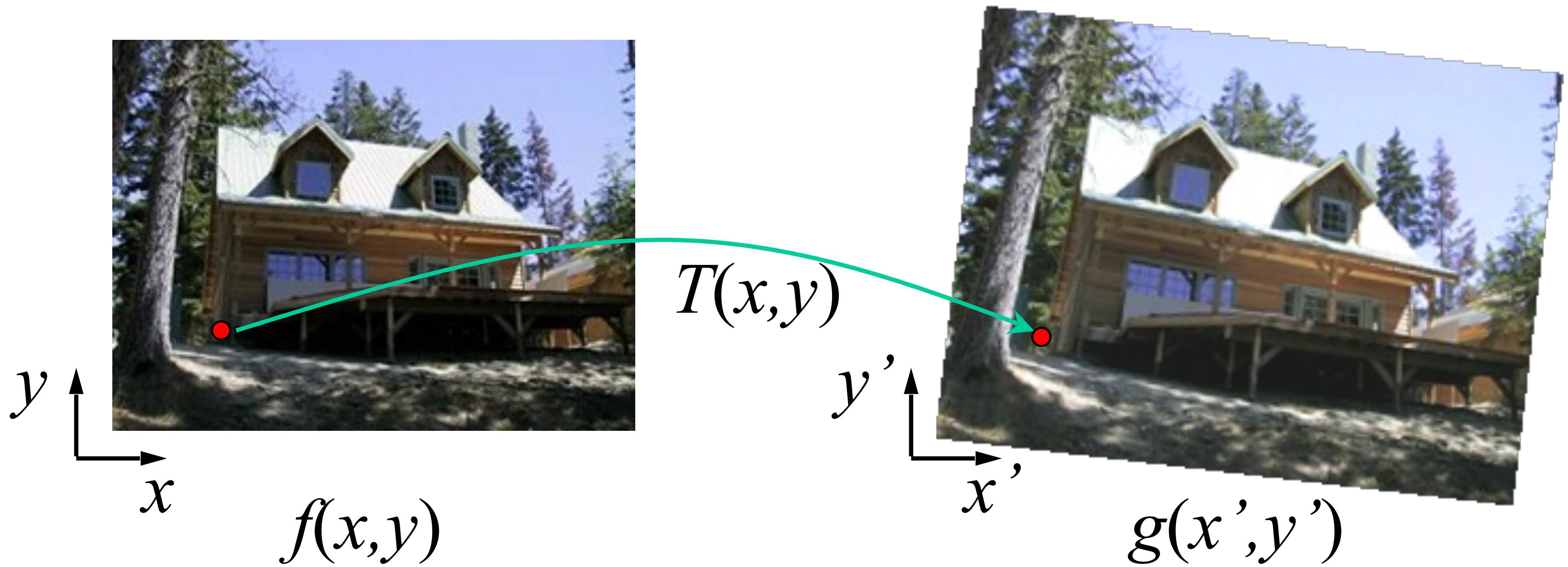
$$T = \begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix}$$

Déformation d'image



- Étant données une image f et une transformation T, comment calculer l'image déformée g?

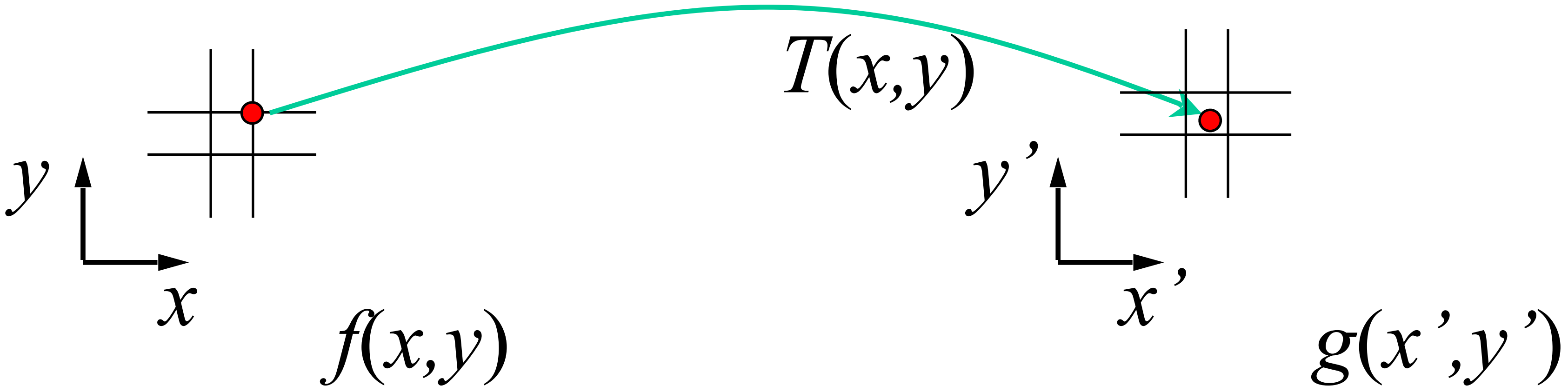
Idée 1: transformée directe



- Pour chaque pixel dans f
  - Calculer sa nouvelle position, et “copier-coller” sa couleur

Idée 1: transformée directe

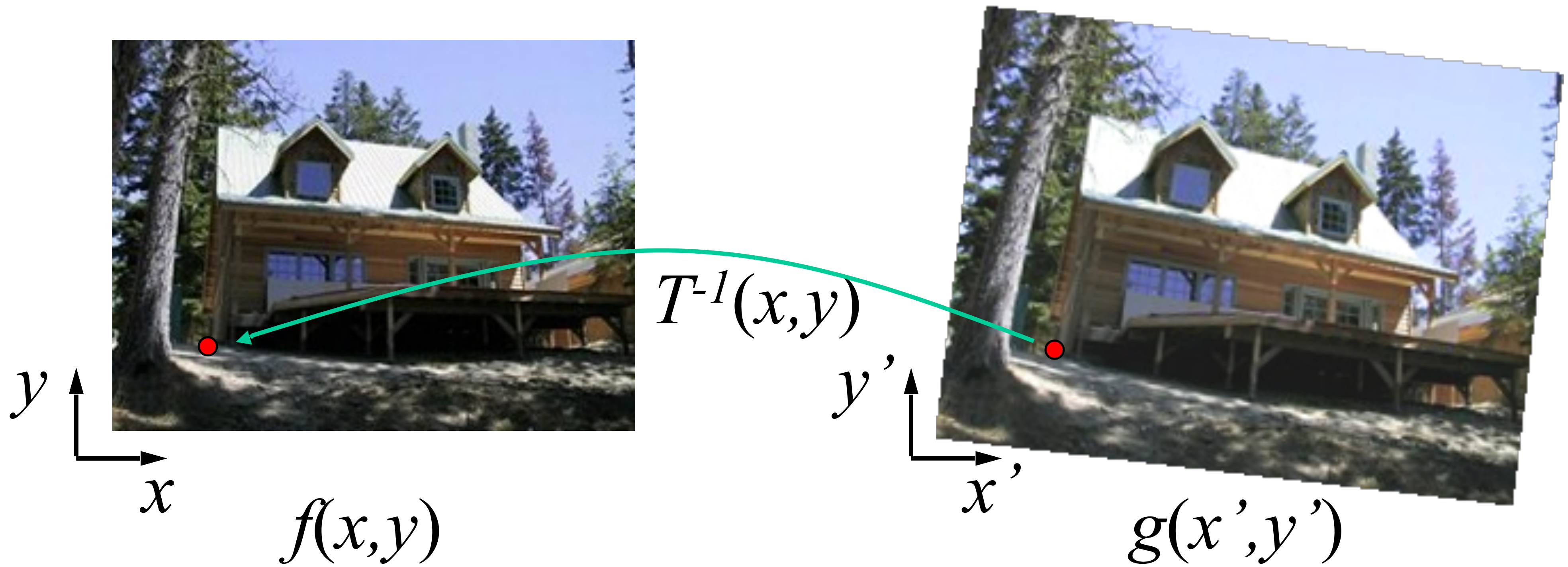
Quel est le problème avec cette approche?



Q: Qu'est-ce qu'on fait si un pixel arrive "entre" deux pixels?

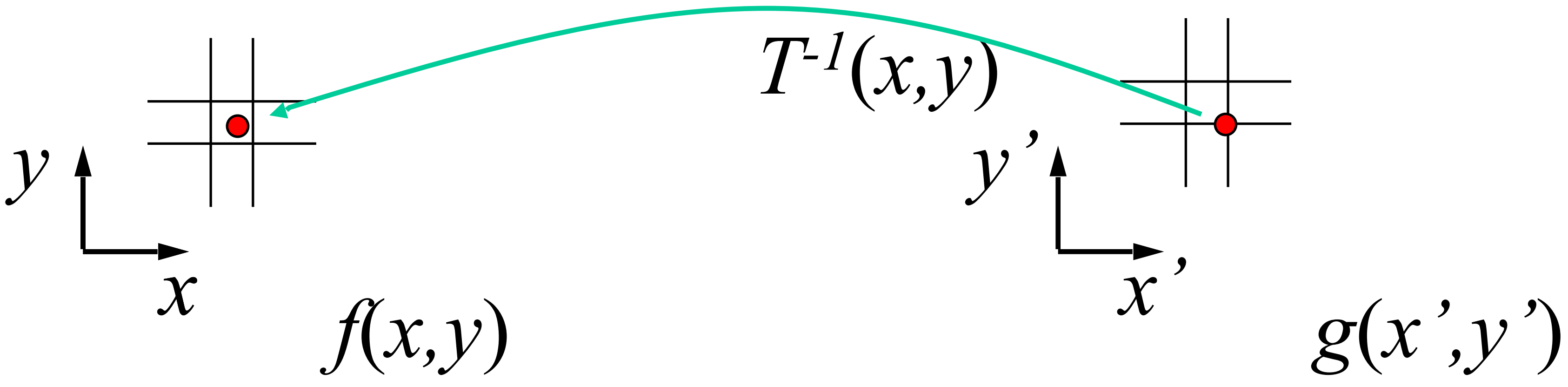
R: distribuer sa couleur sur les pixels avoisinants
(comme si on "aplatissait" la couleur)

Idée 2: transformée inverse



- Pour chaque pixel dans g
  - Calculer d'où il vient grâce à l'inverse de T

Idée 2: transformée inverse

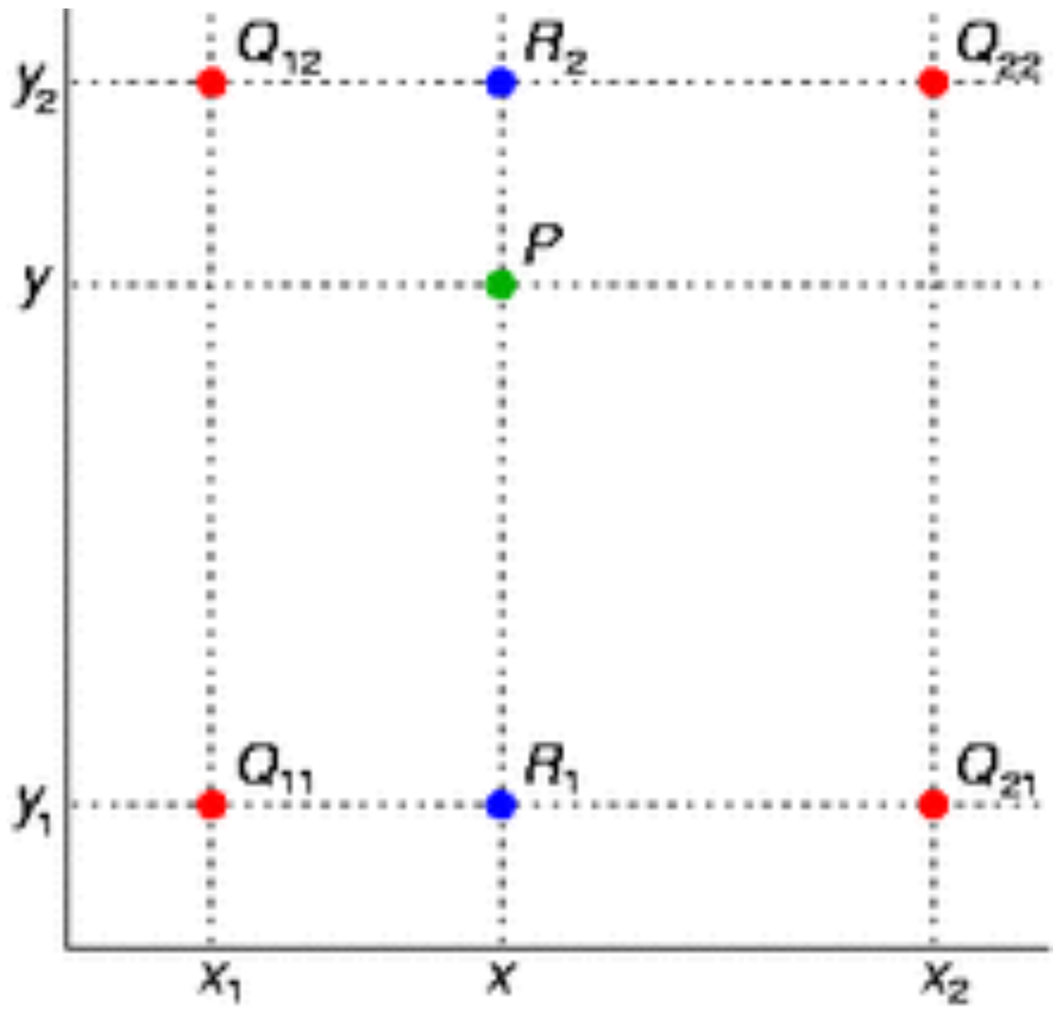


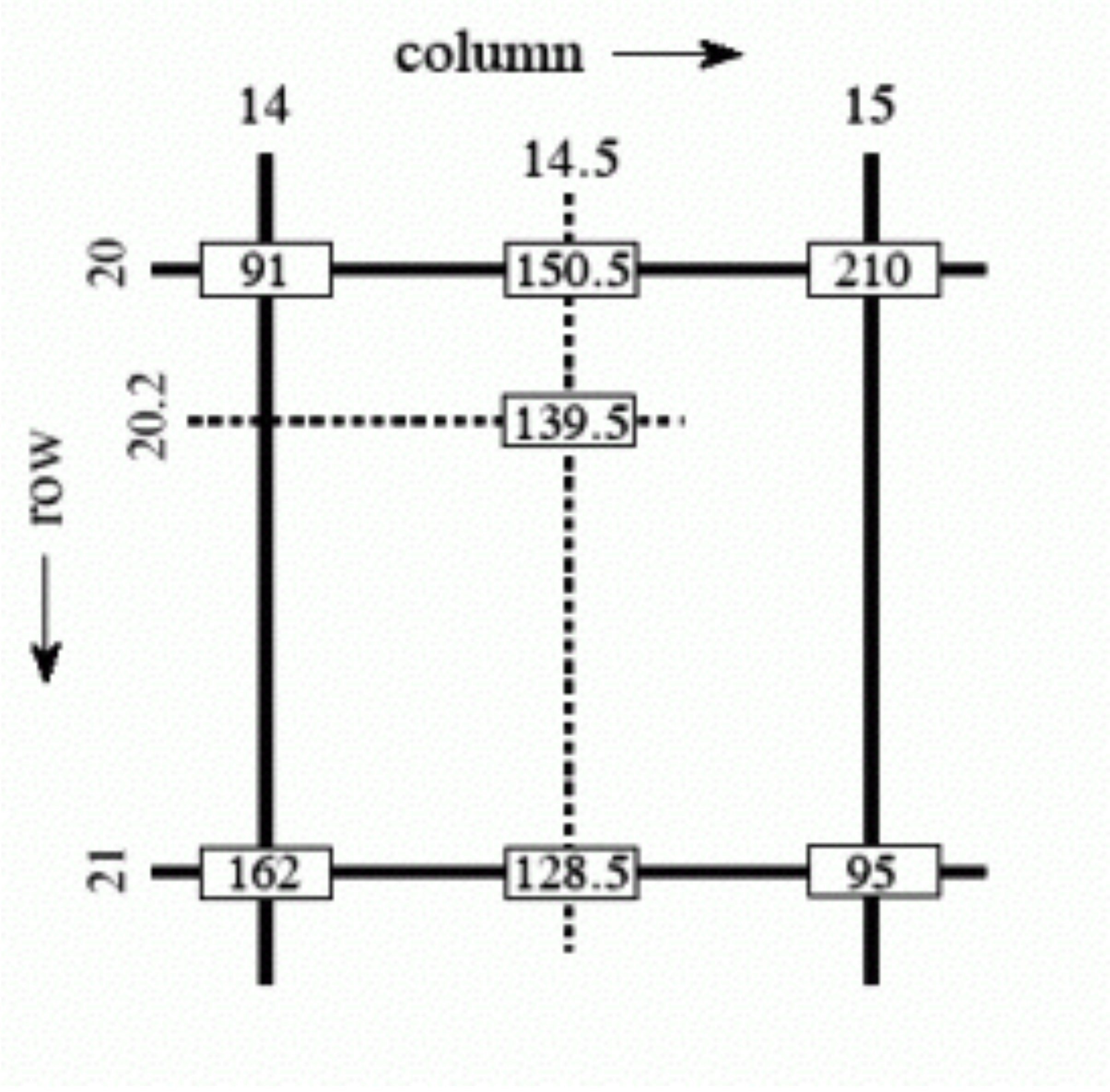
Q: qu'est-ce qu'on fait si un pixel provient "d'entre deux pixels"?

R: Interpolation!
plus proche voisin, bi-linéaire, bi-cubique, etc.
interp2 dans Matlab

Interpolation bilinéaire

$$f(x,y) \approx \begin{bmatrix}1-x & x\end{bmatrix}\begin{bmatrix}f(0,0) & f(0,1)\\ f(1,0) & f(1,1)\end{bmatrix}\begin{bmatrix}1-y\\ y\end{bmatrix}.$$





http://en.wikipedia.org/wiki/Bilinear_interpolation

Déformation directe vs inverse

- Laquelle est la meilleure?
- Habituellement, c'est la transformée inverse
  - Garantit qu'on ne génère pas de trou
  - Cependant, il faut que notre transformation puisse être inversée!