

Outils Informatiques pour le Multimédia

Master 1 informatique
Tronc commun

Département d'informatique
Université Paul Sabatier

Extraction de l'information à partir d'images

- 1 Introduction et définitions
- 2 Opérations arithmétiques
- 3 Opérations spatiales
- 4 Opérations géométriques
 - Transformation en coordonnées homogènes
 - Algorithmes de transformation géométriques
- 5 Exemples d'applications de traitement d'images

Extraction de l'information à partir d'images

Introduction

- Objectifs des applications de traitement d'images
 - Corriger et préparer une image pour son affichage.
 - Détecter la présence ou l'absence d'un objet.
 - Calculer les caractéristiques d'un élément de l'image.
 - ...
- Points communs entre les applications :
 - Extraction d'informations à partir d'images

Extraction de l'information à partir d'images

Introduction

- Objectifs des applications de traitement d'images
 - Corriger et préparer une image pour son affichage.
 - Détecter la présence ou l'absence d'un objet.
 - Calculer les caractéristiques d'un élément de l'image.
 - ...
- Points communs entre les applications :
 - **Extraction d'informations à partir d'images**

Opérateurs de traitement d'images

- Briques logicielles pouvant être combinées et enchaînées pour identifier ou isoler l'information recherchée.

Définitions

Opérateurs de traitement d'images

- Analogie avec les opérateurs mathématiques :
 - Opération prenant en entrée une image ou un ensemble d'informations relatif à une image, et produisant une image ou un ensemble d'informations relatives aux données initiales.

Définitions

Opérateurs de traitement d'images

- Analogie avec les opérateurs mathématiques :
 - Opération prenant en entrée une image ou un ensemble d'informations relatif à une image, et produisant une image ou un ensemble d'informations relatives aux données initiales.
- Analogie similaire avec les programmes informatiques

Définitions

Opérateurs de traitement d'images

- Analogie avec les opérateurs mathématiques :
 - Opération prenant en entrée une image ou un ensemble d'informations relatif à une image, et produisant une image ou un ensemble d'informations relatives aux données initiales.
- Analogie similaire avec les programmes informatiques
- Familles d'opérateurs
 - Image \rightarrow Image
 - Image \rightarrow Informations
 - Informations \rightarrow Images

Opérateurs de traitement d'images

- Opérateurs Image \rightarrow Image
 - Pixel à pixel :
arithmétique, couleur, dynamique ...
 - Locaux
flou, morphologie, contours ...
 - Fréquentiels
réduction de bruit, filtres passe-bande ...
 - Globaux
calcul de distance ...

Opérateurs de traitement d'images

- Opérateurs Image → Informations
 - Segmentation en frontière, en région.
découpage de l'images en zone ...
 - Classification de pixels.
construction d'une relation pixel/information ...
 - Calcul de paramètres.
paramètres intrinsèques ou extrinsèques de la caméra ...

Opérateurs de traitement d'images

- Informations → Images
 - Visualisation de données
météo, mécanique, finance ...
 - Dessin vectoriel.
création artistique, illustration
 - Opérateurs de rendu.
synthèse d'images

Opérateurs de ~~traitement~~ synthèse d'images

- Informations → Images
 - Visualisation de données
météo, mécanique, finance ...
 - Dessin vectoriel.
création artistique, illustration
 - Opérateurs de rendu.
synthèse d'images

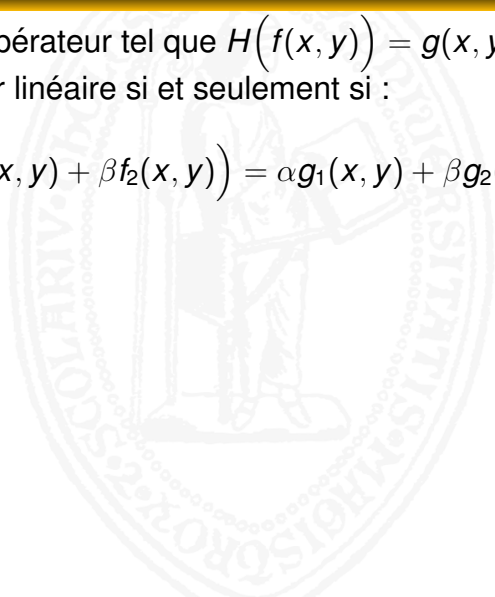
Opérateurs de traitement d'images

Un bon opérateur, c'est quoi ? ? ?

Définition

Soit H , un opérateur tel que $H(f(x, y)) = g(x, y)$. H est un opérateur linéaire si et seulement si :

$$H(\alpha f_1(x, y) + \beta f_2(x, y)) = \alpha g_1(x, y) + \beta g_2(x, y)$$



Définition

Soit H , un opérateur tel que $H(f(x, y)) = g(x, y)$. H est un opérateur linéaire si et seulement si :

$$H(\alpha f_1(x, y) + \beta f_2(x, y)) = \alpha g_1(x, y) + \beta g_2(x, y)$$

Exemples

- $H(I(x, y)) = 2I(x, y)$ est linéaire

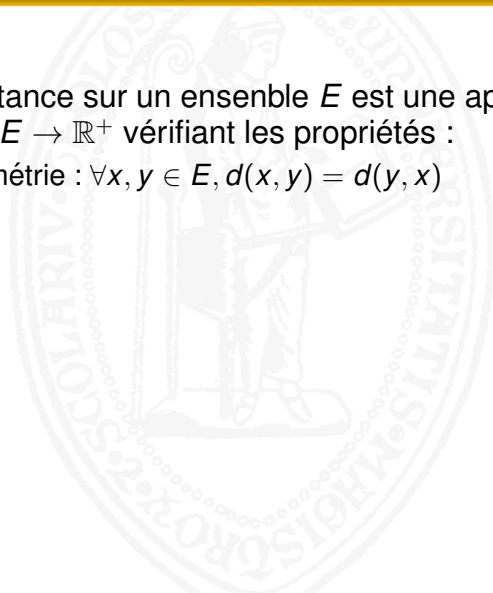
$$\begin{aligned} H(\alpha I(x, y) + \beta J(x, y)) &= 2(\alpha I(x, y) + \beta J(x, y)) \\ &= \alpha \times 2I(x, y) + \beta \times 2J(x, y) \\ &= \alpha H(I(x, y)) + \beta H(J(x, y)) \end{aligned}$$

- $H(I(x, y)) = \max_{(x, y)}(I)$ n'est pas linéaire

Définitions

Distance

- Une distance sur un ensemble E est une application $d : E \times E \rightarrow \mathbb{R}^+$ vérifiant les propriétés :
 - Symétrie : $\forall x, y \in E, d(x, y) = d(y, x)$



Définitions

Distance

- Une distance sur un ensemble E est une application $d : E \times E \rightarrow \mathbb{R}^+$ vérifiant les propriétés :
 - Symétrie : $\forall x, y \in E, d(x, y) = d(y, x)$
 - Séparation : $\forall x, y \in E, d(x, y) = 0 \Leftrightarrow x = y$

Définitions

Distance

- Une distance sur un ensemble E est une application $d : E \times E \rightarrow \mathbb{R}^+$ vérifiant les propriétés :
 - Symétrie : $\forall x, y \in E, d(x, y) = d(y, x)$
 - Séparation : $\forall x, y \in E, d(x, y) = 0 \Leftrightarrow x = y$
 - Inégalité triangulaire :
 $\forall x, y, z \in E, d(x, z) \leq d(x, y) + d(y, z)$

Définitions

Distance

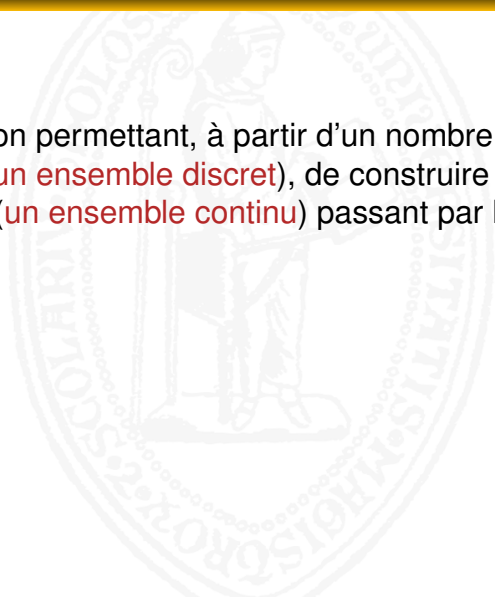
- Une distance sur un ensemble E est une application $d : E \times E \rightarrow \mathbb{R}^+$ vérifiant les propriétés :
 - Symétrie : $\forall x, y \in E, d(x, y) = d(y, x)$
 - Séparation : $\forall x, y \in E, d(x, y) = 0 \Leftrightarrow x = y$
 - Inégalité triangulaire :

$$\forall x, y, z \in E, d(x, z) \leq d(x, y) + d(y, z)$$
- Distances usuelles
 - Manhattan (1-distance) : $\sum_{i=1}^d |x_i - y_i|$
 - Euclidienne (2-distance) : $\sqrt{\sum_{i=1}^d |x_i - y_i|^2}$
 - Minkowski (p-distance) : $\sqrt[p]{\sum_{i=1}^d |x_i - y_i|^p}$
 - Tchebychev (∞ -distance) : $\max_{1 \leq i \leq d} |x_i - y_i|$

Définitions

Interpolation

- Opération permettant, à partir d'un nombre fini de points (**un ensemble discret**), de construire une courbe (**un ensemble continu**) passant par les points initiaux.



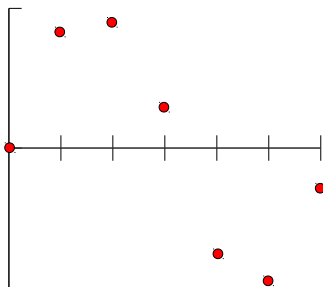
Définitions

Interpolation

- Opération permettant, à partir d'un nombre fini de points (**un ensemble discret**), de construire une courbe (**un ensemble continu**) passant par les points initiaux.
- Applications :
 - Re-échantillonnage
 - Reconstruction
 - Lissage
 - ...

Interpolation

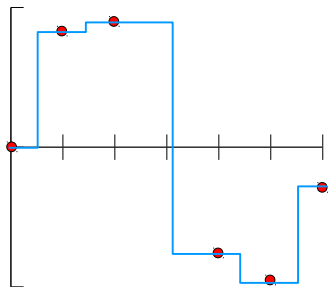
- Fonction $f(x)$, discrétisée en 7 points $f(x_i)$ avec $x_i \in E = \{0, 1, 2, 3, 4, 5, 6\}$



- Combien vaut $f(1.3)$?

Interpolation au plus proche voisin

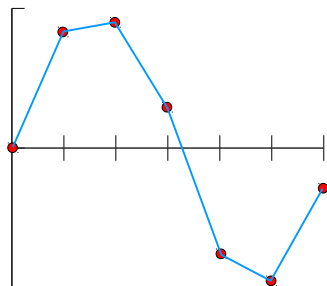
- Soit fonction $ppv(x)$ retournant le point $x_i \in E$ le plus proche de x
- $ppv(x) = round(x)$



- $f(1.3) = f(ppv(1.3)) = f(1)$
- Reconstruction constante par morceaux

Interpolation linéaire

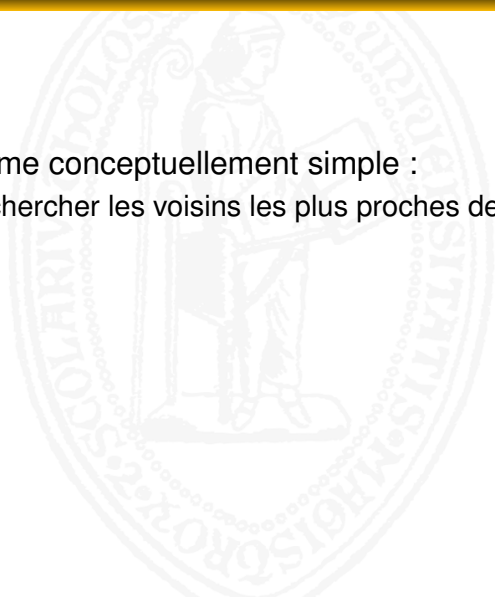
- Soit la droite $L_{x_i}^{x_{i+1}}(x)$ passant par les points x_i et x_{i+1} de E encadrant x
- $L_{x_i}^{x_{i+1}}(x) = a(x - x_i) + b$ avec $a = \frac{y_{i+1} - y_i}{x_{i+1} - x_i}$ et $b = y_i$



- $f(1.3) = L_1^2(1.3)$
- Reconstruction linéaire par morceaux

Interpolation dans la pratique

- Algorithme conceptuellement simple :
 - Rechercher les voisins les plus proches de x



Interpolation dans la pratique

- Algorithme conceptuellement simple :
 - Rechercher les voisins les plus proches de x
 - Evaluer l'équation d'interpolation

Interpolation dans la pratique

- Algorithme conceptuellement simple :
 - Rechercher les voisins les plus proches de x
 - Evaluer l'équation d'interpolation
 - Attention à la complexité de recherche
 - Utilisation d'un arbre de recherche : *kd-tree*

Interpolation dans la pratique

- Algorithme conceptuellement simple :
 - Rechercher les voisins les plus proches de x
 - Evaluer l'équation d'interpolation
 - Attention à la complexité de recherche
 - Utilisation d'un arbre de recherche : *kd-tree*
 - Programmation directe mais efficace

Opérateurs de traitement d'images

Ça, c'est la base ! Et on fait quoi avec ?

Opérations arithmétiques

- Opérateurs pixels à pixels :
 - Somme : $s(x, y) = f(x, y) + g(x, y)$
 - Différence : $d(x, y) = f(x, y) - g(x, y)$
 - Produit : $p(x, y) = f(x, y) * g(x, y)$
 - Quotient : $q(x, y) = f(x, y) / g(x, y)$

Exemple : Produit de 2 images 2x2

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \times \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} = \begin{bmatrix} a_{11} * b_{11} & a_{12} * b_{12} \\ a_{21} * b_{21} & a_{22} * b_{22} \end{bmatrix}$$

Opérations arithmétiques

- Opérateurs pixels à pixels :
 - Somme : $s(x, y) = f(x, y) + g(x, y)$
 - Différence : $d(x, y) = f(x, y) - g(x, y)$
 - Produit : $p(x, y) = f(x, y) * g(x, y)$
 - Quotient : $q(x, y) = f(x, y) / g(x, y)$

Exemple : Produit de 2 images 2x2

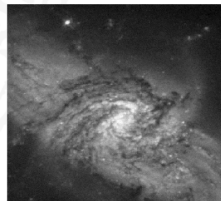
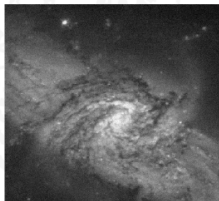
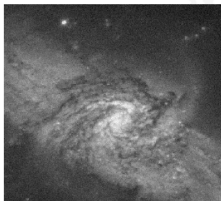
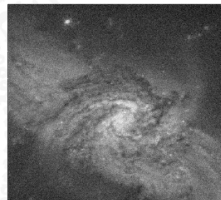
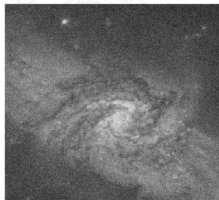
$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \times \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} = \begin{bmatrix} a_{11} * b_{11} & a_{12} * b_{12} \\ a_{21} * b_{21} & a_{22} * b_{22} \end{bmatrix}$$

- Implémenté par parcours pixel à pixel des images
- Parallélisation triviale

Opérations arithmétiques

Exemples

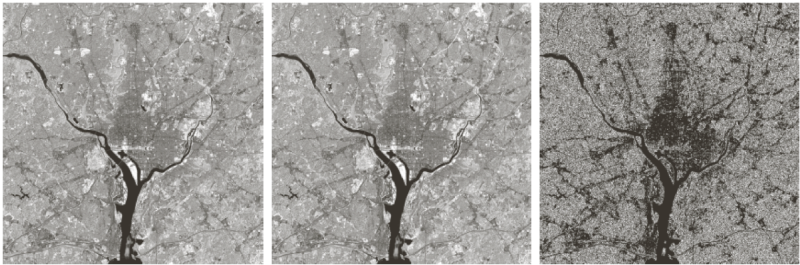
Moyenne d'images bruitées → débruitage



Opérations arithmétiques

Exemples

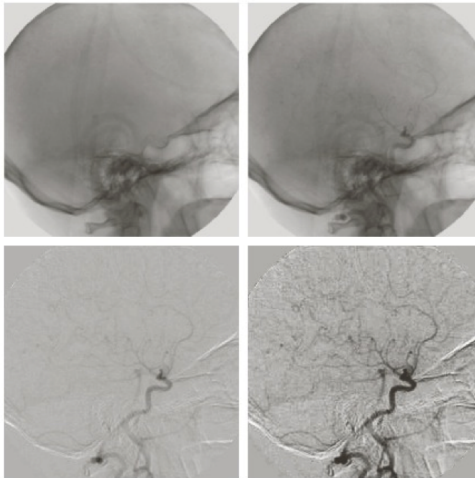
Soustraction d'images → évolution



Opérations arithmétiques

Exemples

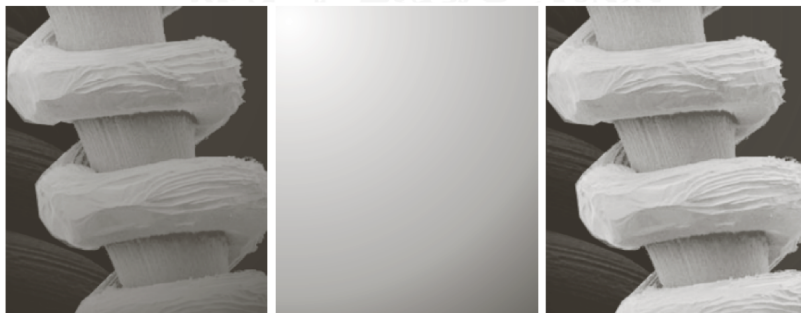
Soustraction après rehaussement → vaisseaux



Opérations arithmétiques

Exemples

Multiplication d'images → correction éclairage



Opérations arithmétiques

Précautions à prendre dans le cas d'opérations arithmétiques

- Division par 0 : rajouter une petite valeur à tous les pixels nuls
- Dépassement
 - On met à 255 toutes les valeurs qui dépassent 255
 - On recadre l'image entre 0 et K (si 8 bits, K = 255)

$$f_m = f - \min(f)$$

$$f_s = K \frac{f_m}{\max(f_m)}$$

Opérateurs de traitement d'images

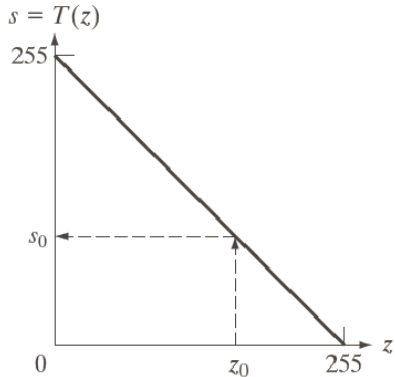
Après le pixel, le groupe de pixels

Opérations spatiales

Modification de la valeur de plusieurs pixels d'une image

- Sans tenir compte des autres pixels

$$s = T(z)$$



Opérations spatiales

Modification de la valeur de plusieurs pixels d'une image

- Sans tenir compte des autres pixels

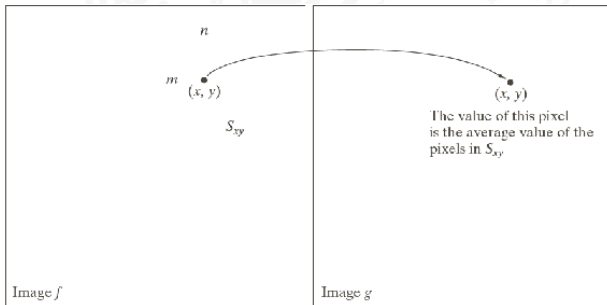


Opérations spatiales

Modification de la valeur de plusieurs pixels d'une image

- En tenant compte des autres pixels
- Opération sur un voisinage

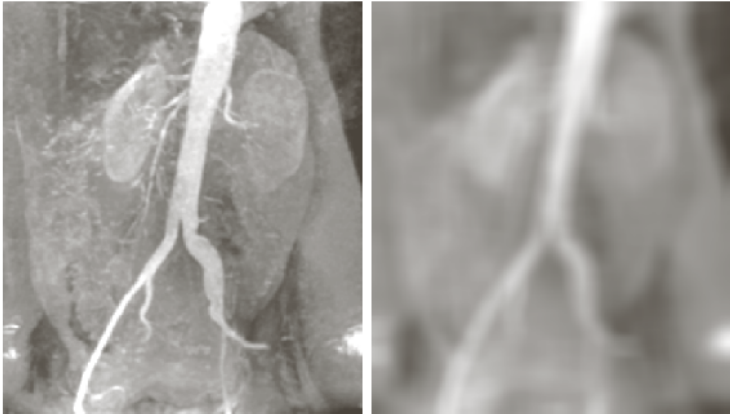
$$g(x, y) = \frac{1}{mn} \sum_{(r,c) \in V_{xy}} f(r, c)$$



Opérations spatiales

Modification de la valeur de plusieurs pixels d'une image

- Elimination des petites structures, filtrage

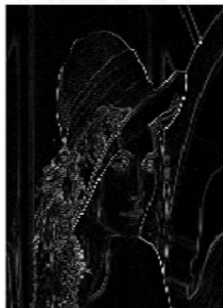


Opérations spatiales

Modification de la valeur de plusieurs pixels d'une image

- Moyenne pondérée : détection de contours

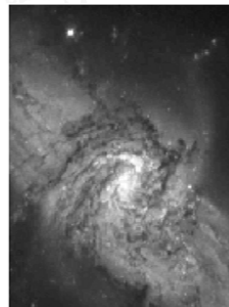
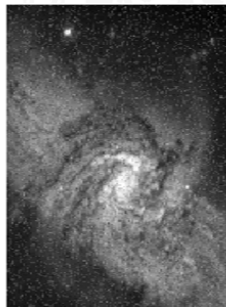
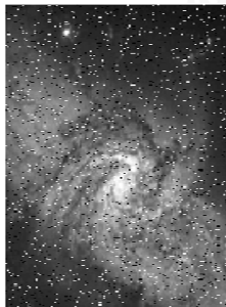
$$f_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \quad g_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$



Opérations spatiales

Modification de la valeur de plusieurs pixels d'une image

- Filtrage non linéaire
 - Calcul de la médiane sur un voisinage
 - Meilleur filtrage de bruit poivre et sel



Opérateurs de traitement d'images

Et si on modifiait la position d'un pixel ?

Opérations géométriques

Modification de la relation spatiale entre les pixels

- Homothéties
- Rotations
- Translations
- Cisaillements
- ...

Homothéties 2D

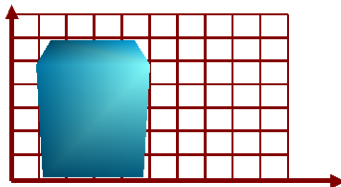
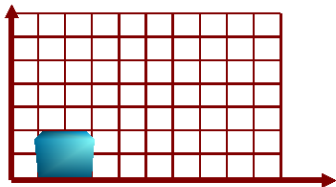
- Mise à l'échelle

$$\begin{cases} x' = s_x \cdot x \\ y' = s_y \cdot y \end{cases}$$

- Notation matricielle

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

- Homothétie isomorphe si $s_x = s_y$



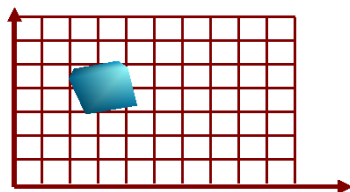
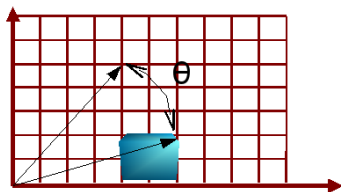
Rotations 2D

- Rotation de centre O

$$\begin{cases} x' = \cos(\theta) \cdot x - \sin(\theta) \cdot y \\ y' = \sin(\theta) \cdot x + \cos(\theta) \cdot y \end{cases}$$

- Notation matricielle

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



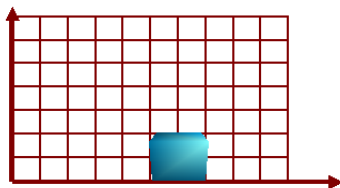
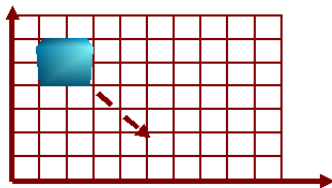
Translation 2D

- Déplacement dans le plan

$$\begin{cases} x' = x + t_x \\ y' = y + t_y \end{cases}$$

- Notation matricielle

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} ? & ? \\ ? & ? \end{bmatrix} \times \begin{bmatrix} x \\ y \end{bmatrix}$$



Translation 2D

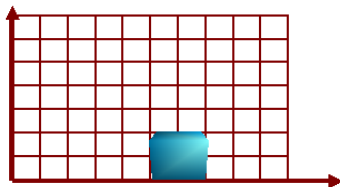
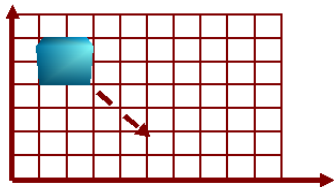
- Déplacement dans le plan

$$\begin{cases} x' = x + t_x \\ y' = y + t_y \end{cases}$$

- Notation matricielle

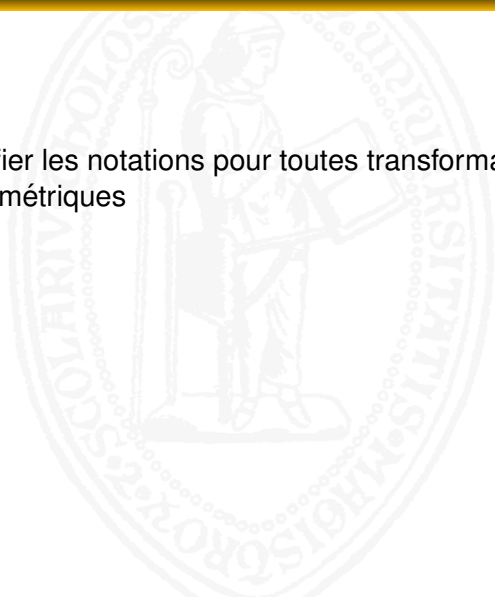
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} ? & ? \\ ? & ? \end{bmatrix} \times \begin{bmatrix} x \\ y \end{bmatrix}$$

Pb : pas de notation matricielle : difficultés d'unification des transformations



Coordonnées homogènes

- Besoins
 - Unifier les notations pour toutes transformations géométriques



Coordonnées homogènes

- Besoins
 - Unifier les notations pour toutes transformations géométriques
 - Combiner les transformation par simple produit matriciel

Coordonnées homogènes

- Besoins
 - Unifier les notations pour toutes transformations géométriques
 - Combiner les transformation par simple produit matriciel
 - Traiter la translation comme une transformation matricielle

Coordonnées homogènes

- Besoins
 - Unifier les notations pour toutes transformations géométriques
 - Combiner les transformation par simple produit matriciel
 - Traiter la translation comme une transformation matricielle
- Solution : définition d'un espace *homogène* de dimension $d + 1$.

Coordonnées homogènes

- Exemple en 2D

- Point 2D en coordonnées homogènes $\begin{bmatrix} x \\ y \\ w \end{bmatrix}$
- Correspond au point en coordonnées cartésiennes $\begin{bmatrix} \frac{x}{w} \\ \frac{y}{w} \end{bmatrix}$ si $w \neq 0$
- Point à l'infini dans la direction $\begin{bmatrix} x \\ y \end{bmatrix}$ sinon

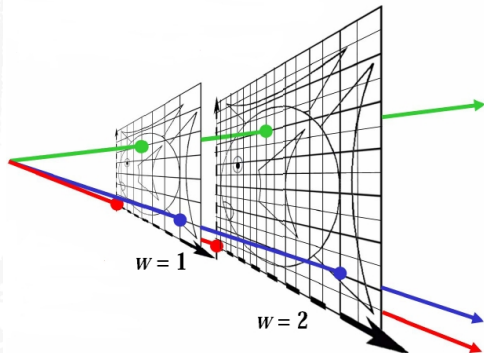
- Matrices de transformation

- Taille 3×3 en 2D
- Taille 4×4 en 3D

Normalisation homogène

- Normalisation : division par w

- $(0, 0, 1) = (0, 0, 2) = \dots$
- $(7, 1, 1) = (14, 2, 2) = \dots$
- $(4, 5, 1) = (8, 10, 2) = \dots$



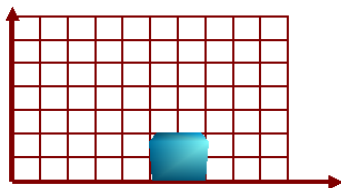
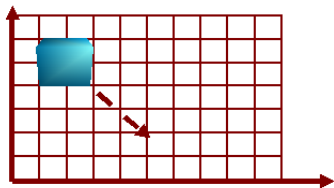
Translation 2D

- Déplacement dans le plan

$$\begin{cases} x' = x + t_x \\ y' = y + t_y \end{cases}$$

- Notation matricielle

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



Opérations géométriques

Transformations les plus utilisées

Identity

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

$$\begin{aligned} x &= x \\ y &= y \end{aligned}$$



Scaling

$$\begin{bmatrix} c_x & 0 & 0 \\ 0 & c_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{aligned} x &= c_x x \\ y &= c_y y \end{aligned}$$



Rotation

$$\begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{aligned} x &= x \cos \theta - y \sin \theta \\ y &= x \sin \theta + y \cos \theta \end{aligned}$$



Translation

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ t_x & t_y & 1 \end{bmatrix}$$

$$\begin{aligned} x &= x + t_x \\ y &= y + t_y \end{aligned}$$



Shear (vertical)

$$\begin{bmatrix} 1 & 0 & 0 \\ s_y & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{aligned} x &= x + s_y y \\ y &= y \end{aligned}$$



Shear (horizontal)

$$\begin{bmatrix} 1 & s_x & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{aligned} x &= x + s_x y \\ y &= y \end{aligned}$$



Opérations géométriques

Et ça se code comment ?

Opérations géométriques

Implantation pour des données discrètes

Approche directe :

Pour chaque pixel (x, y) de l'image source **faire**

$$(x', y') = \text{Arrondi}\left(\text{Transformation}(x, y)\right)$$

$$\text{imageDestination}[x', y'] = \text{imageSource}[x, y]$$

Fin

Opérations géométriques

Implantation pour des données discrètes

Approche directe :

Pour chaque pixel (x, y) de l'image source **faire**

$$(x', y') = \text{Arrondi}\left(\text{Transformation}(x, y)\right)$$

$$\text{imageDestination}[x', y'] = \text{imageSource}[x, y]$$

Fin

- **Problèmes liés aux arrondis !**

- Deux pixels (x_0, y_0) et (x_1, y_1) peuvent donner la même destination (x', y')
- Perte d'information, trous, ...

Opérations géométriques

Implantation pour des données discrètes

Approche inverse :

Pour chaque pixel (x, y) de l'image de destination
faire

$$(x', y') = \text{TransformationInverse}(x, y)$$
$$\text{imageDestination}[x, y] =$$
$$\text{Interpolation}(\text{imageSource}, (x', y'))$$

Fin

Opérations géométriques

Implantation pour des données discrètes

Approche inverse :

Pour chaque pixel (x, y) de l'image de destination
faire

$$\begin{aligned}(x', y') &= \text{TransformationInverse}(x, y) \\ \text{imageDestination}[x, y] &= \\ &\text{Interpolation}\left(\text{imageSource}, (x', y')\right)\end{aligned}$$

Fin

- Interpolation = re-échantillonnage, reconstruction
 - Plus proches voisins : génère des à-plats
 - Linéaire : lissage de l'image

Opérations géométriques

Plus proche voisin versus bilinéaire



Opérations géométriques

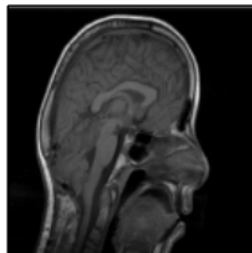
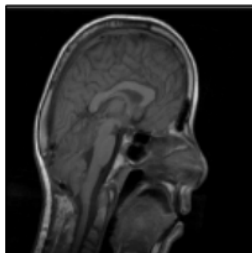
A quoi ça sert ?

Opérations géométriques

Applications

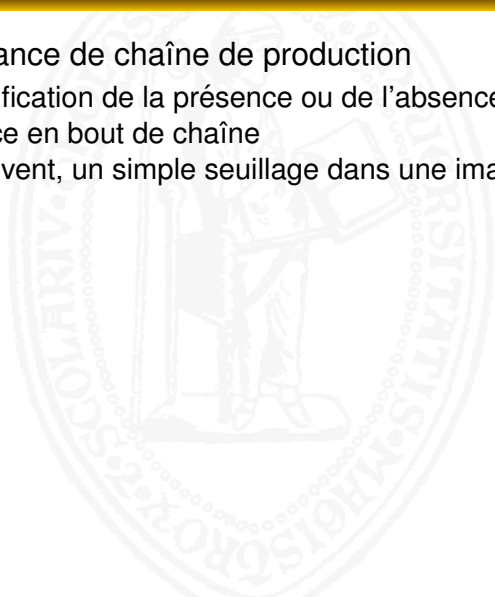
Recalage d'images

- Deux images d'une même scène
- Deux points de vue différents
- Et si on souhaite les faire correspondre ?



Exemples concrets

- Surveillance de chaîne de production
 - Vérification de la présence ou de l'absence d'une pièce en bout de chaîne
 - Souvent, un simple seuillage dans une image



Exemples concrets

- Surveillance de chaîne de production
 - Vérification de la présence ou de l'absence d'une pièce en bout de chaîne
 - Souvent, un simple seuillage dans une image
- Contrôle de la maturité en agriculture
 - Reconnaissance de la couleur et de la texture d'un fruit, légume, ...
 - Segmentation et classification

Exemples concrets

- Surveillance de chaîne de production
 - Vérification de la présence ou de l'absence d'une pièce en bout de chaîne
 - Souvent, un simple seuillage dans une image
- Contrôle de la maturité en agriculture
 - Reconnaissance de la couleur et de la texture d'un fruit, légume, ...
 - Segmentation et classification
- Construction et correction de cartes géographiques
 - Recalage et simplification de photographies aériennes
 - Transformation géométriques et filtrage

Exemples concrets

- Segmentation et suivi de cellules vivantes en microscopie
 - Détection d'anomalies biologiques et médicales
 - Segmentation et classification et transformation géométriques

Exemples concrets

- Segmentation et suivi de cellules vivantes en microscopie
 - Détection d'anomalies biologiques et médicales
 - Segmentation et classification et transformation géométriques
- Analyse et authentification de tableaux
 - Etudes des couleurs et textures
 - Segmentation, classification et comparaison

Exemples concrets

- Segmentation et suivi de cellules vivantes en microscopie
 - Détection d'anomalies biologiques et médicales
 - Segmentation et classification et transformation géométriques
- Analyse et authentification de tableaux
 - Etudes des couleurs et textures
 - Segmentation, classification et comparaison
- Recherche d'image par le contenu
 - Etudes des formes, couleurs, texture, ...
 - Segmentation, classification et transformation