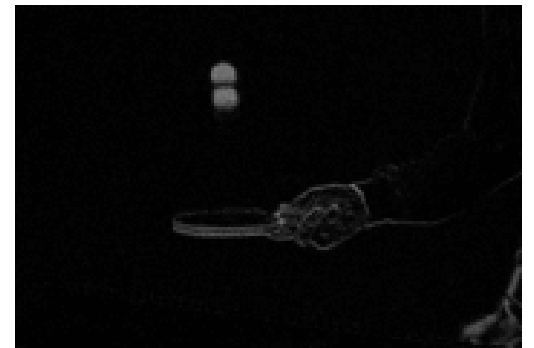
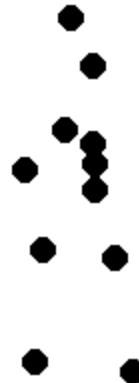
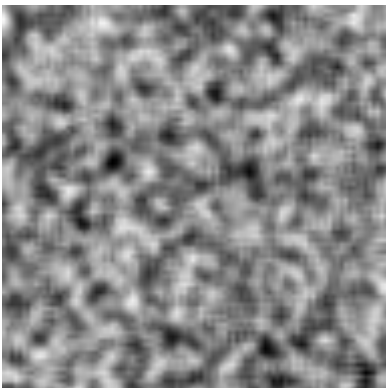


Analyse du mouvement

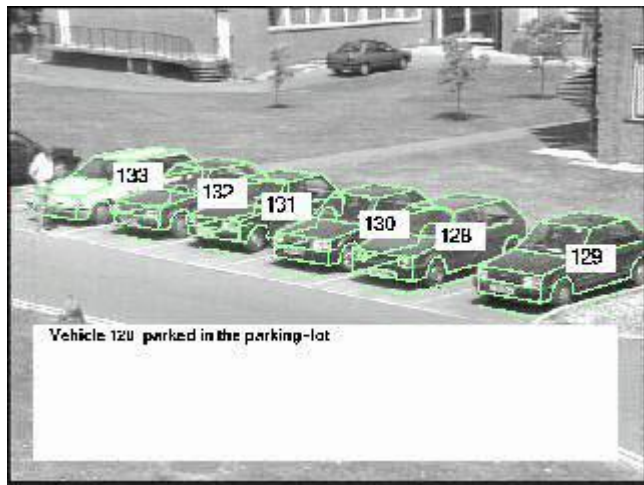
Estimation & Poursuite

Antoine MANZANERA – ENSTA/LEI



Applications et enjeux (1)

Surveillance de zone



(University of Surrey)

Compression et Indexation



Applications et enjeux (2)

Navigation robotique



(Ecole des Mines / CMM)

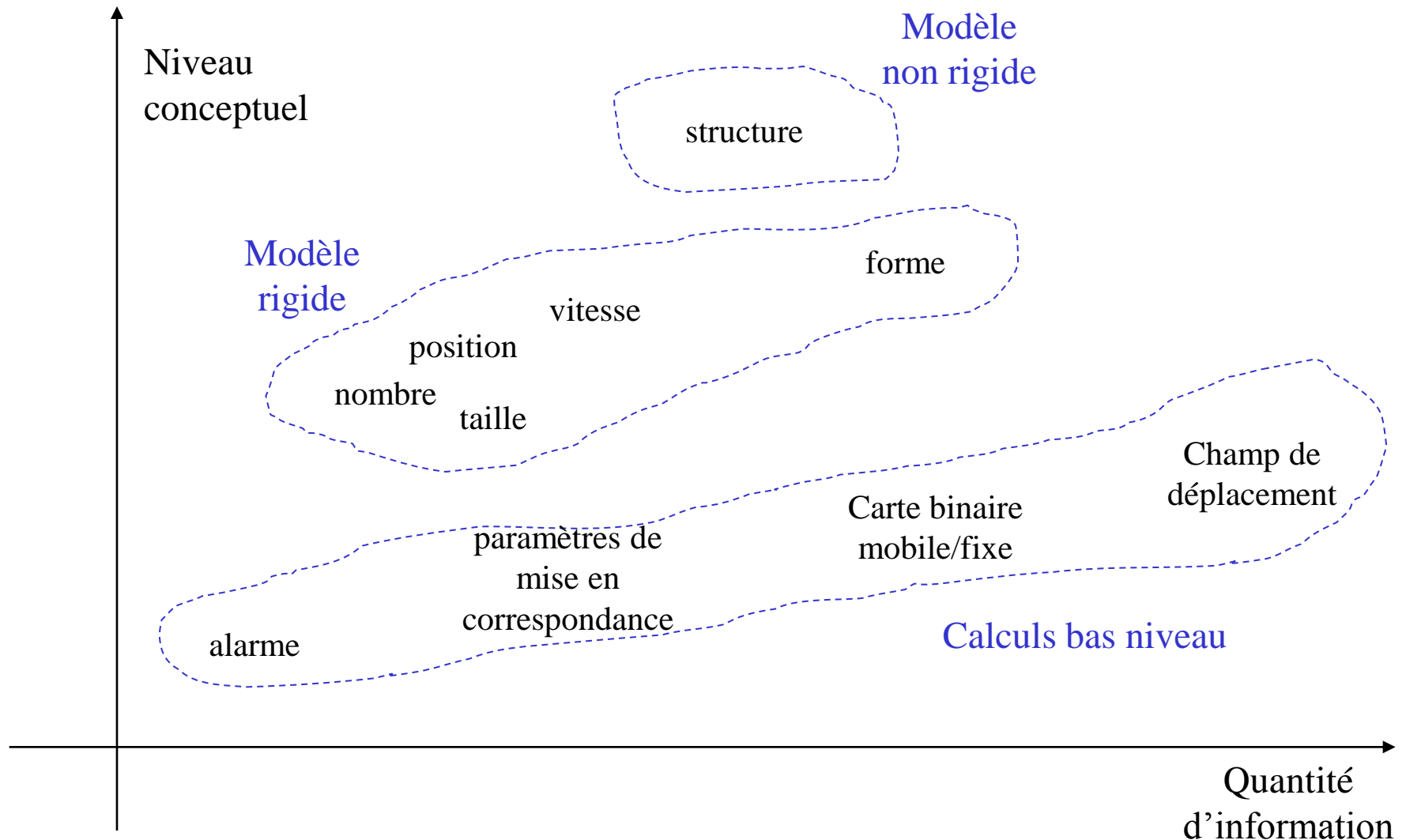
Poursuite automatique



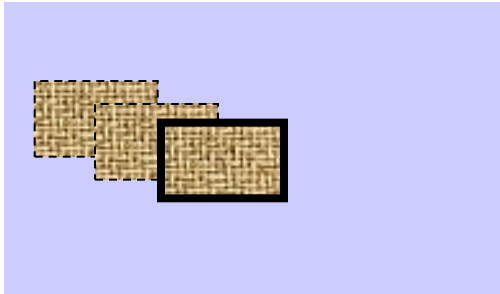
Mouvement fluide



Quelle information extraire ?



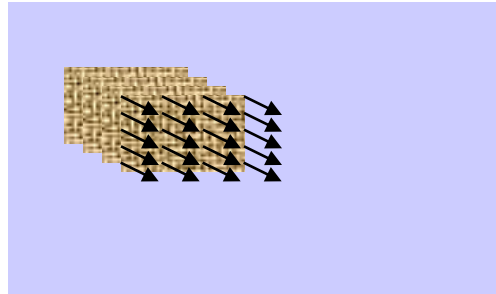
Détection – Estimation – Poursuite



DETECTION

Objectif : identifier dans chaque image les pixels appartenant à des objets mobiles

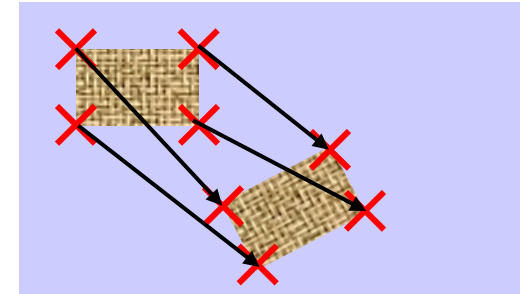
- Certaine continuité temporelle
- Mouvement de la caméra nul ou très simple.



ESTIMATION

Objectif : calculer le mouvement apparent (vitesse instantanée) de chaque pixel

- Continuité temporelle
- Plutôt « traiter après »



POURSUITE

Objectif : apparier certaines structures spatiales pour chaque couple d'images.

- Discontinuité temporelle
- Plutôt « traiter avant »



Chapitre 1



Chapitre 2

PLAN DU CHAPITRE

- **Champ de vitesse apparent et applications**
- **Techniques par appariement**
- **Techniques différentielles 1 : Lucas et Kanade**
- **Techniques différentielles 2 : Horn et Schunck**
- **Technique du plus proche voisin projeté**

Calcul du mouvement apparent

(1) Le calcul d'un mouvement apparent *global* (mise en correspondance) entre deux images correspond à l'estimation des paramètres d'une transformation affectant *tous* les points de l'image : translation, rotation, homothétie, affinité,...

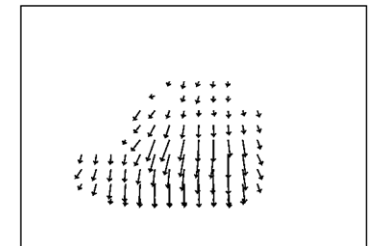
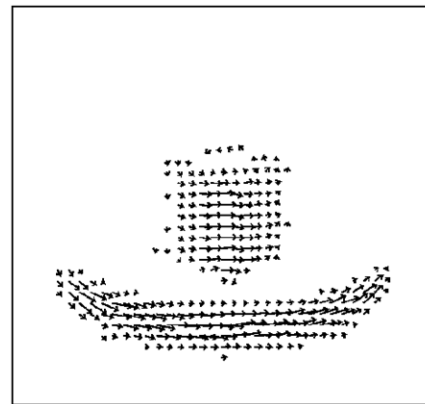
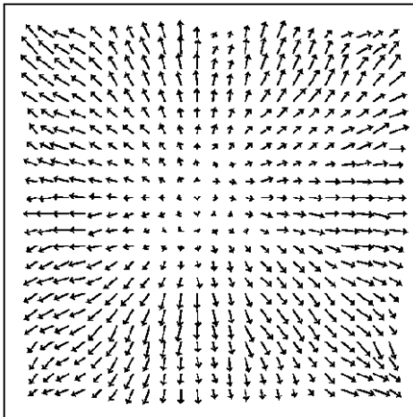
(2) Le calcul du mouvement apparent *local* consiste à associer à chaque pixel (x,y,t) de I un vecteur (v_x^t, v_y^t) représentant la *vitesse apparente* du pixel (x,y) à l'instant t .

—————→ Calcul du *flot optique* (= Champ de mouvement apparent)

Idéalement : le vecteur (v_x^t, v_y^t) représente la projection sur le plan image du vecteur vitesse (V_x^t, V_y^t, V_z^t) des objets de la scène par rapport au repère image (O,x,y,z) (grandeur objective).

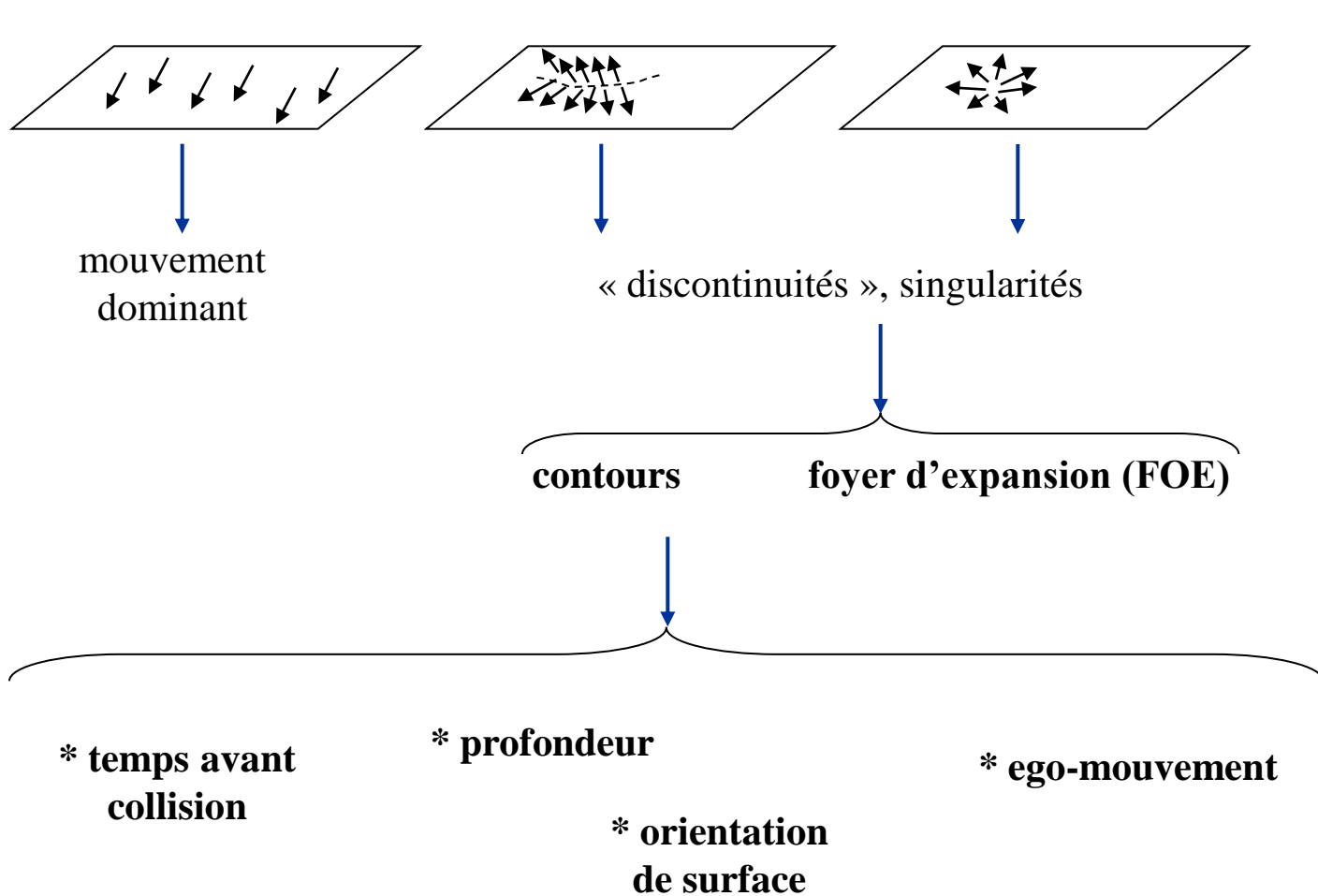
On le calcule à partir des variations temporelles de la fonction $I(x,y,t)$.

Exemples de flot optique



Source : Pierre Kornprobst - INRIA

Quelles informations peut fournir le flot optique ?



niveau
conceptuel

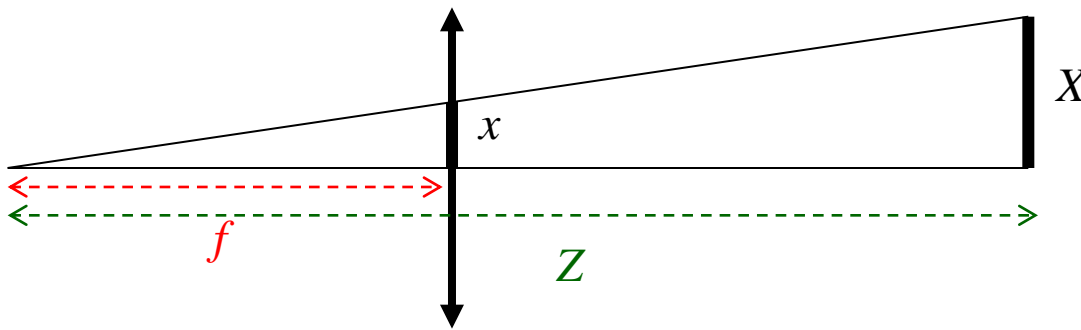
Distortion perspective

On note (X,Y,Z) les coordonnées d'un point de la scène.

(x,y) les coordonnées du même point projeté dans l'image

f la distance focale de la caméra

Distortion perspective
(*modèle sténopé*) :



$$x = f \frac{X}{Z}$$
$$y = f \frac{Y}{Z}$$

Mouvement projectif

Supposons que la caméra se déplace à la vitesse $(-X', -Y', -Z')$ par rapport à une scène statique (scène non déformable, pas d'objets mobiles), alors tous les points de la scène sont à la même vitesse (X', Y', Z') par rapport à la caméra (avec $X' = dX/dt$; $Y' = dY/dt$; $Z' = dZ/dt$). En dérivant par rapport au temps des équations de distorsion perspective :

$$\left\{ \begin{array}{l} x' = f \left(\frac{X'}{Z} - \frac{XZ'}{Z^2} \right) \\ y' = f \left(\frac{Y'}{Z} - \frac{YZ'}{Z^2} \right) \end{array} \right. \quad \text{Soit :} \quad \begin{pmatrix} x' \\ y' \end{pmatrix} = \frac{1}{Z} \begin{pmatrix} -f & 0 & x \\ 0 & -f & y \end{pmatrix} \begin{pmatrix} -X' \\ -Y' \\ -Z' \end{pmatrix}$$

Un modèle plus réaliste...

En réalité, le centre optique (projection du diaphragme sur le plan image) n'est jamais au centre de l'image mais en un point (x_c, y_c) . De plus, la distortion perspective n'est jamais la même dans les deux directions, et la distance focale est remplacée par le couple (f_x, f_y) :

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \frac{1}{Z} \begin{pmatrix} -f_x & 0 & (x - x_c) \\ 0 & -f_y & (y - y_c) \end{pmatrix} \begin{pmatrix} -X' \\ -Y' \\ -Z' \end{pmatrix}$$

Les paramètres $\{x_c, y_c, f_x, f_y\}$ dépendent du capteur et des paramètres de l'optique. Ils sont estimés dans une phase de *calibration* de la caméra.

D'autre part, si on prend en compte la composante de rotation $\Omega = (\Omega_x, \Omega_y, \Omega_z)$ de la caméra par rapport à la scène :

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \frac{1}{Z} \begin{pmatrix} -f_x & 0 \\ 0 & -f_y \end{pmatrix} \begin{pmatrix} x - x_c \\ y - y_c \end{pmatrix} + \begin{pmatrix} \frac{(x-x_c)(y-y_c)}{f_y} & f_y + \frac{(y-y_c)^2}{f_y} & \frac{f_x}{f_y}(y-y_c) \\ f_y + \frac{(y-y_c)^2}{f_y} & -\frac{(x-x_c)(y-y_c)}{f_x} & -(x-x_c) \end{pmatrix} \begin{pmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{pmatrix}$$

facteurs de perspective
coordonnées du centre optique
composante de translation
composante de rotation

Translation pure et profondeur

- Translation pure de la caméra selon l'axe de x , soit $T = (-X', 0, 0)$ et $\Omega = (0, 0, 0)$

$$\begin{cases} x' = f_x \frac{X'}{Z} \\ y' = 0 \end{cases} \quad \text{soit} \quad \begin{cases} Z = f_x \frac{X'}{x'} \end{cases}$$

————→ Vitesses apparentes horizontales, de modules inversement proportionnels à la profondeur.

- Translation pure de la caméra selon l'axe de z , soit $T = (0, 0, -Z')$ et $\Omega = (0, 0, 0)$

$$\begin{cases} x' = \frac{(x - x_c)Z'}{Z} \\ y' = \frac{(y - y_c)Z'}{Z} \end{cases} \quad \text{soit} \quad \begin{cases} Z = \frac{(x - x_c)Z'}{x'} \\ Z = \frac{(y - y_c)Z'}{y'} \end{cases}$$

————→ Zoom sur l'image avec un foyer d'expansion au niveau du centre optique.

Le foyer d'expansion (FOE)

Lors d'une translation de la caméra dans une scène statique, les directions de vitesse des points projetés sur le plan image converge vers un point du plan projectif appelé FOE.

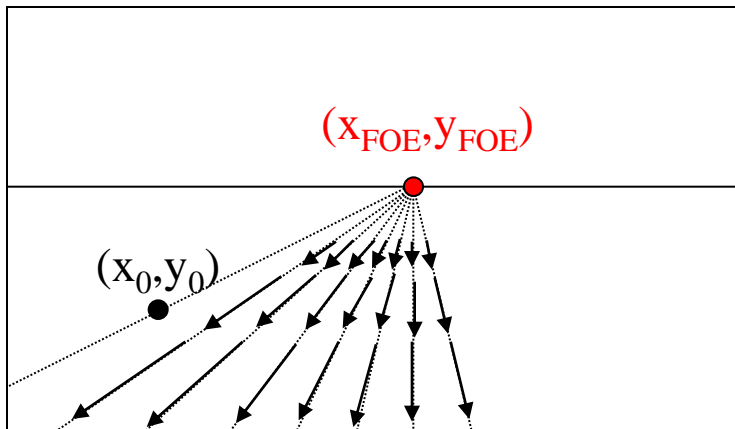
Dans la suite, on suppose que la caméra se déplace selon la translation $T = (-X', -Y', -Z')$. Pour simplifier les notations, on suppose de plus que $f_x = f_y = f$ et $(x_c, y_c) = (0, 0)$.

Soit (X_0, Y_0, Z_0) un point de la scène. Après un temps t , il est projeté sur l'image au point (x_t, y_t) , avec :

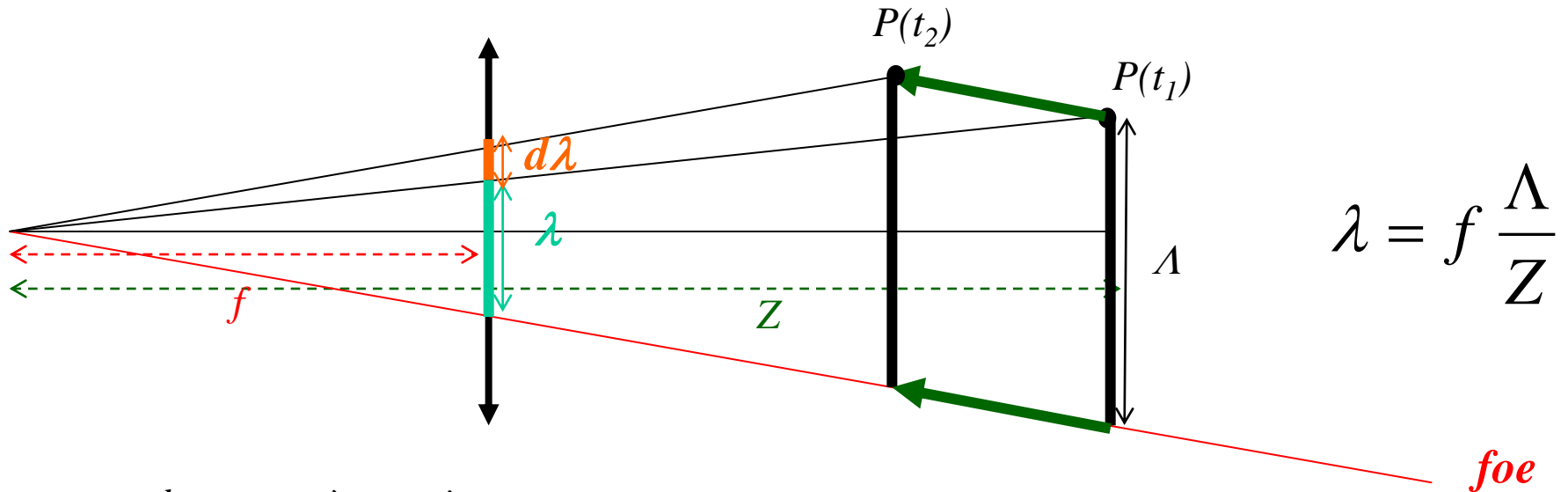
$$(x_t, y_t) = \left(f \frac{X_0 + tX'}{Z_0 + tZ'}, f \frac{Y_0 + tY'}{Z_0 + tZ'} \right)$$

$$(x_{FOE}, y_{FOE}) = \lim_{t \rightarrow -\infty} (x_t, y_t)$$

$$= \left(f \frac{X'}{Z'}, f \frac{Y'}{Z'} \right)$$



Temps avant collision



mouvement dans une scène statique :

$$\lambda' = -\frac{f\Lambda}{Z^2} Z'$$

et donc

$$\frac{\lambda}{\lambda'} = -\frac{Z}{Z'}$$

temps avant
collision

Calcul du flot optique : limites et contraintes

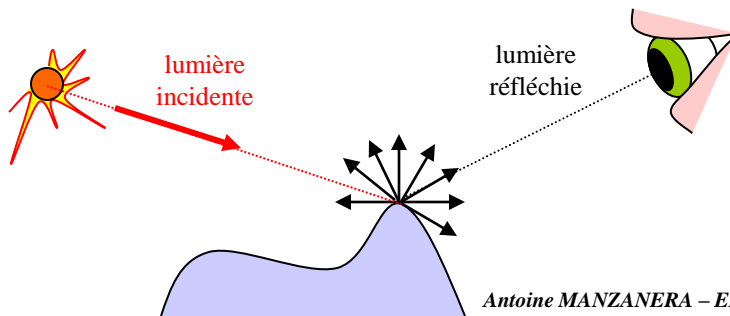


(1) On suppose :

MOUVEMENT \Leftrightarrow VARIATION D'INTENSITE

(\Leftarrow) On suppose que l'observation d'une variation d'intensité dans l'image traduit nécessairement l'existence d'un mouvement dans la scène. Cela correspond à une hypothèse *d'éclairement constant* sur la scène.

(\Rightarrow) C'est la variation de l'intensité dans l'image qui doit permettre de retrouver le mouvement apparent de l'objet dans la scène. On suppose donc que la lumière réfléchie par un point de la scène reste constante indépendamment du mouvement relatif scène / caméra. L'hypothèse sous-jacente est que les objets sont à surface *lambertienne* :



L'intensité de la lumière réfléchie est la même dans toutes les directions.

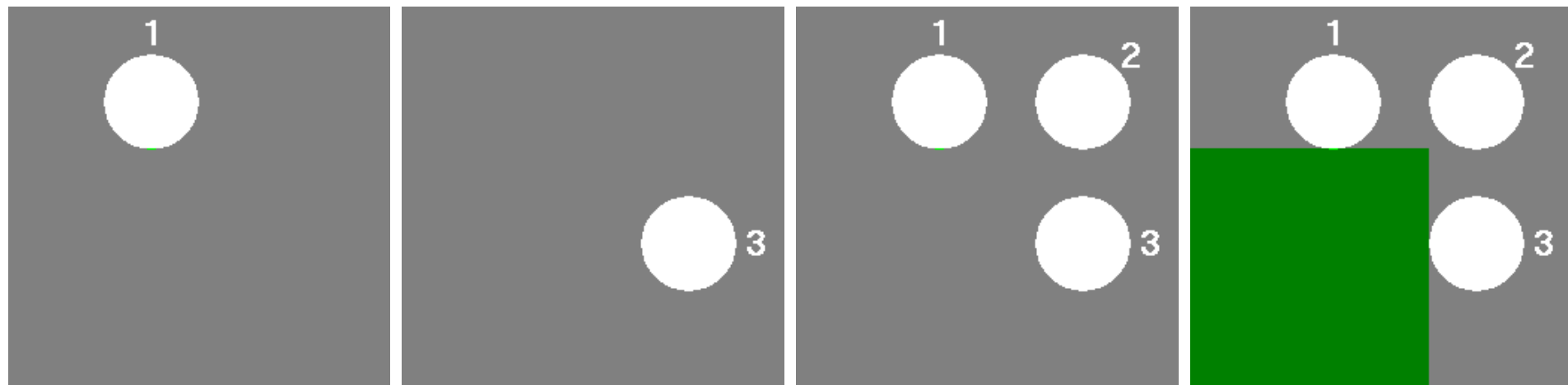
Calcul du flot optique : limites et contraintes



(2) *PROBLEME DE L'OUVERTURE*

On n' « accède » au mouvement apparent d'un point que grâce à un calcul effectué *dans un voisinage borné* de ce point.

➡ On ne peut calculer que la composante du mouvement *dans la direction du gradient* (i.e. *perpendiculaire au contour*).

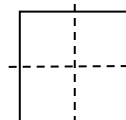


Source : PROJET TELESUN <http://telesun.insa-lyon.fr>

Technique par appariement

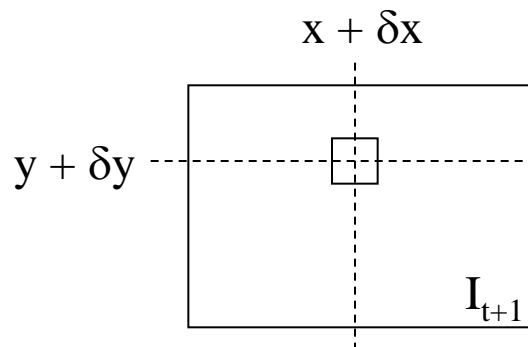
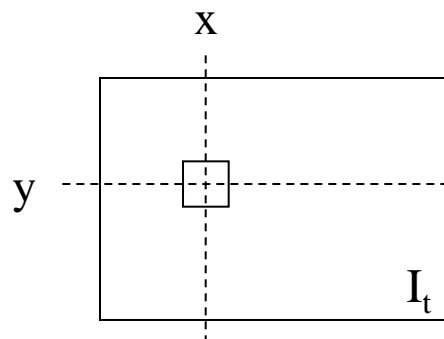
Mesure d'appariement :

On considère $B \subset \mathbb{Z}^2$ voisinage de l'origine :



$$A_{(x,y)}^t(\delta x, \delta y) = \sum_{(b_1, b_2) \in B} (I(x + b_1, y + b_2, t) - I(x + \delta x + b_1, y + \delta y + b_2, t + 1))^2$$

C'est la somme des différences au carrée (SSD) de 2 blocs :



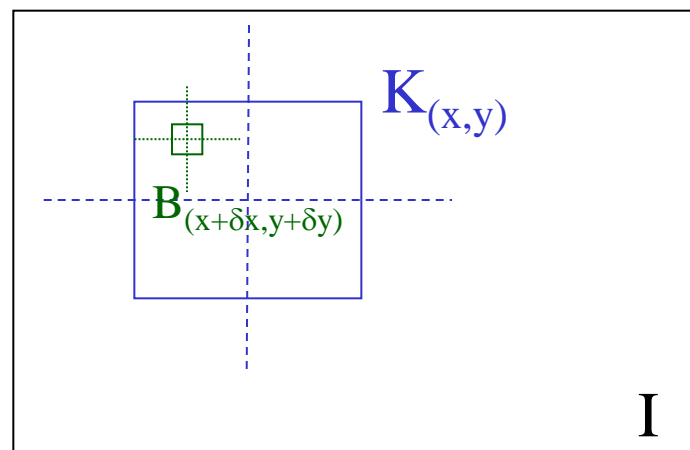
typiquement :
B de taille
9×9, 15×15...

Solution du flot optique :

$$(v_x^t, v_y^t) = \arg \min_{(\delta x, \delta y) \in K} A_{(x,y)}^t(\delta x, \delta y)$$

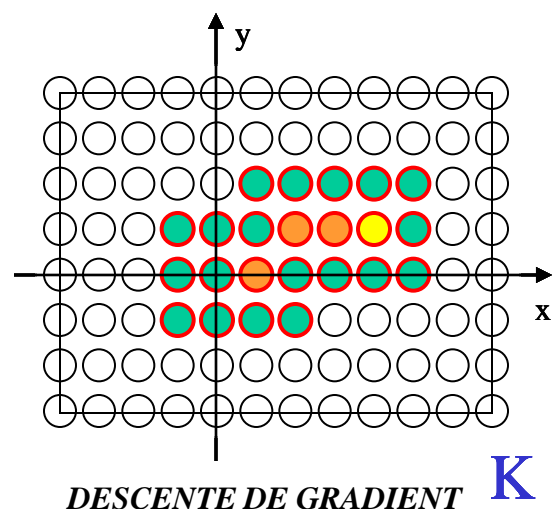
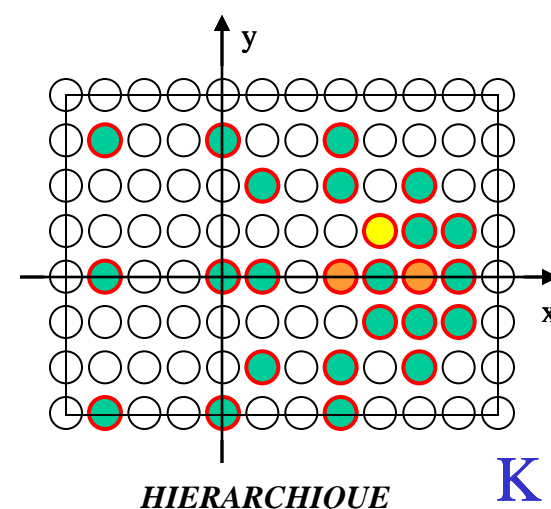
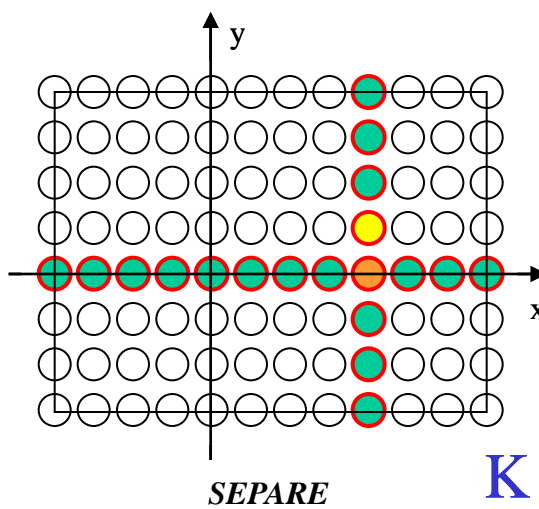
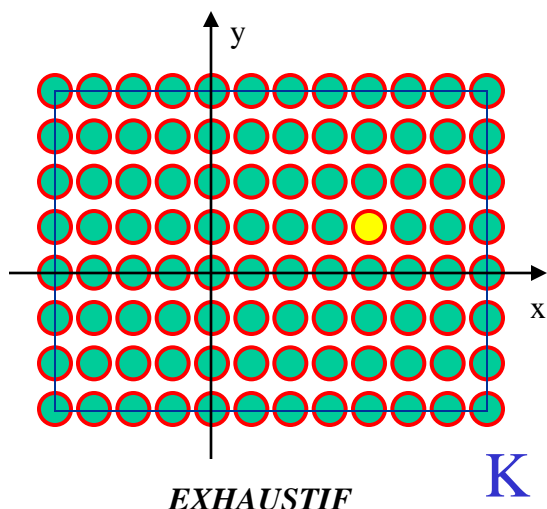
$$K \subsetneq I$$

pour limiter le temps de calcul
+ stratégies d'optimisation...



Appariement : stratégies d'optimisation

Le principe des stratégies d'optimisation est de parcourir « intelligemment » l'espace de recherche K pour minimiser la complexité de calcul de l'optimum d'appariement :



Pour chaque point de K , on ne calcule la somme des différences que tant qu'on demeure inférieur à un seuil T . Le meilleur déplacement correspond à celui pour lequel le plus de points ont été examinés.

SEQUENTIEL

- Ces différentes techniques ne sont pas exclusives, et peuvent être combinées.

- En l'absence d'hypothèse sur la fonction d'appariement (séparabilité, convexité...) dans le domaine K , seul le parcours exhaustif garantit le calcul d'un optimum *global* sur K .

Exemple : normes mpeg de codage video



Flot optique sous-résolu

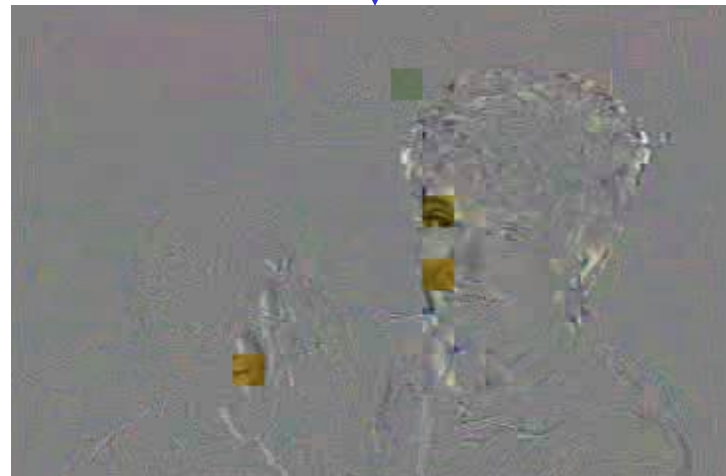


Trame prédite

- Découpage en blocs 16×16 ou 32×32 .
- Hypothèse de même déplacement pour tous les pixels du blocs.
- Codage : déplacements + erreurs comprimées.

↳ Compression statique *JPEG* :

- Transformée en cosinus discrète
- Quantification
- Codage entropique



Erreurs de prédiction

Référence :

VcDemo
TU Delft

Techniques différentielles (1)

LUCAS & KANADE 1981

Le principe de la méthode de Lucas et Kanade est de calculer le minimum de la fonction d'appariement quadratique (SSD), en supposant que le déplacement recherché est petit, comme le point où les dérivées de la fonction d'appariement s'annulent, par rapport à δ_x et à δ_y .

$$\begin{aligned}(v_x^t, v_y^t) &= \arg \min_{(\delta x, \delta y) \in K} A(\delta x, \delta y) \\ &= \arg \min_{(\delta x, \delta y) \in K} \sum_{(x, y) \in B} (I(x, y, t) - I(x + \delta x, y + \delta y, t + 1))^2\end{aligned}$$

Sous l'hypothèse que I est régulière et que le déplacement (δ_x, δ_y) est petit, on écrit le développement de Taylor à l'ordre 1 de I :

$$I(x + \delta x, y + \delta y, t + 1) \approx I(x, y, t) + \frac{\partial I}{\partial x} \delta x + \frac{\partial I}{\partial y} \delta y + \frac{\partial I}{\partial t}$$

La fonction d'appariement A devient :

$$A(\delta x, \delta y) \approx \sum_{(x, y) \in B} \left(\frac{\partial I}{\partial x} \delta x + \frac{\partial I}{\partial y} \delta y + \frac{\partial I}{\partial t} \right)^2$$

Techniques différentielles (1)

LUCAS & KANADE 1981

$$A(\delta x, \delta y) \approx \sum_{(x,y) \in B} \left(\frac{\partial I}{\partial x} \delta x + \frac{\partial I}{\partial y} \delta y + \frac{\partial I}{\partial t} \right)^2$$

Equations d'Euler-Lagrange de minimisation de A : Annulation des dérivées premières par rapport à δx et à δy :

$$2 \sum_{(x,y) \in B} \left(\frac{\partial I}{\partial x} \delta x + \frac{\partial I}{\partial y} \delta y + \frac{\partial I}{\partial t} \right) \frac{\partial I}{\partial x} = 0$$

$$2 \sum_{(x,y) \in B} \left(\frac{\partial I}{\partial x} \delta x + \frac{\partial I}{\partial y} \delta y + \frac{\partial I}{\partial t} \right) \frac{\partial I}{\partial y} = 0$$

Ce qui revient à la résolution du système linéaire :

$$\mathbf{H} \cdot \mathbf{v} = \mathbf{b}$$

$$\text{avec } \mathbf{H} = \begin{pmatrix} \sum_{(x,y) \in B} \left(\frac{\partial I}{\partial x} \right)^2 & \sum_{(x,y) \in B} \frac{\partial I}{\partial x} \frac{\partial I}{\partial y} \\ \sum_{(x,y) \in B} \frac{\partial I}{\partial x} \frac{\partial I}{\partial y} & \sum_{(x,y) \in B} \left(\frac{\partial I}{\partial y} \right)^2 \end{pmatrix} \text{ et } \mathbf{b} = \begin{pmatrix} - \sum_{(x,y) \in B} \frac{\partial I}{\partial x} \frac{\partial I}{\partial t} \\ - \sum_{(x,y) \in B} \frac{\partial I}{\partial y} \frac{\partial I}{\partial t} \end{pmatrix}$$

et $\mathbf{v} = (v_x, v_y)$ le déplacement recherché.

La résolution du système (S) est finalement réalisée par une méthode itérative, type Gauss-Seidel :

$$\text{Initialisation : } v_x^{(0)} = 0; v_y^{(0)} = 0$$

$$\text{Pour } k > 0 : \begin{cases} v_x^{(k)} = \frac{1}{H_1^1} (\mathbf{b} - \mathbf{H}_1^2 \cdot v_y^{(k-1)}) \\ v_y^{(k)} = \frac{1}{H_2^2} (\mathbf{b} - \mathbf{H}_2^1 \cdot v_x^{(k)}) \end{cases}$$

Techniques différentielles (1)

LUCAS & KANADE 1981

L'existence et la stabilité d'une solution au système (S) dépend de la matrice \mathbf{H} :

$$\mathbf{H} = \begin{pmatrix} \sum_{(x,y) \in B} \left(\frac{\partial I}{\partial x} \right)^2 & \sum_{(x,y) \in B} \frac{\partial I}{\partial x} \frac{\partial I}{\partial y} \\ \sum_{(x,y) \in B} \frac{\partial I}{\partial x} \frac{\partial I}{\partial y} & \sum_{(x,y) \in B} \left(\frac{\partial I}{\partial y} \right)^2 \end{pmatrix}$$

NB : On retrouvera cette matrice au chapitre 2 pour le calcul des points d'intérêt...

La matrice \mathbf{H} doit être de rang 2 « au sens fort », c'est-à-dire posséder 2 valeurs propres (λ_1, λ_2) *grandes*. Les auteurs proposent d'utiliser $\lambda = \min(\lambda_1, \lambda_2)$ comme indicateur de confiance du déplacement calculé.

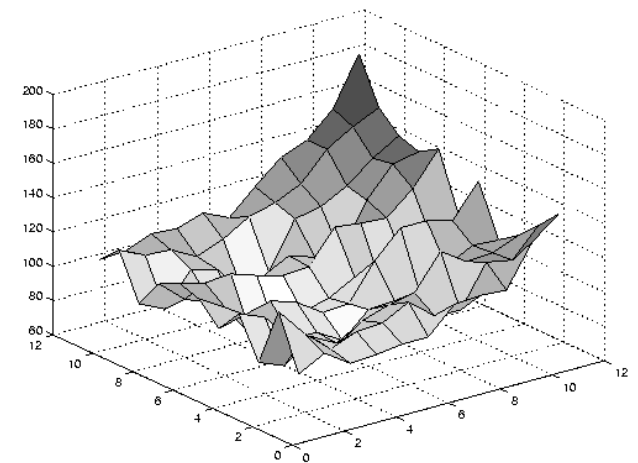
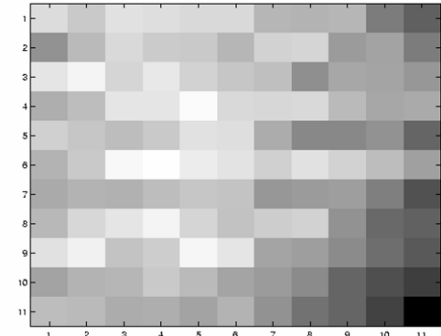
Cette propriété correspond à une interprétation algébrique du *problème de l'ouverture*...

Rang 0 : zone homogène



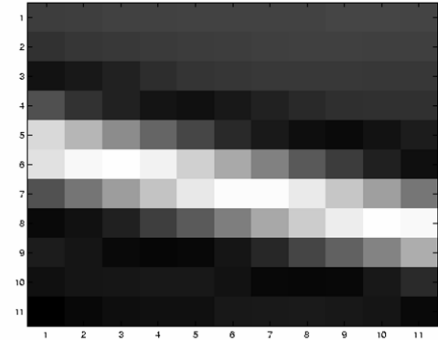
Zone faiblement texturée :
2 valeurs propres faibles

Sources : Steven Seitz UW-Seattle

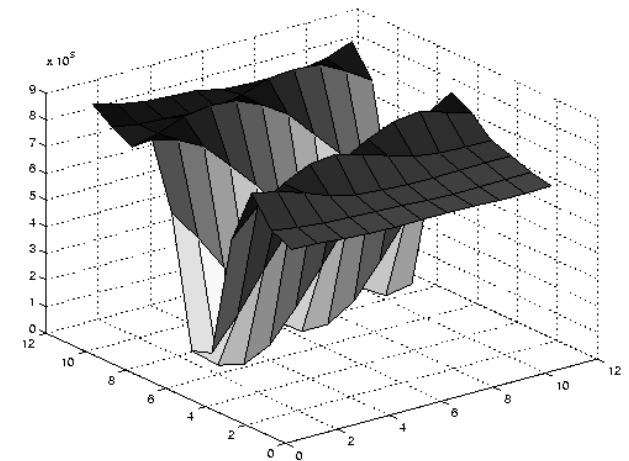


Fonction d'appariement A

Rang 1 : contour simple



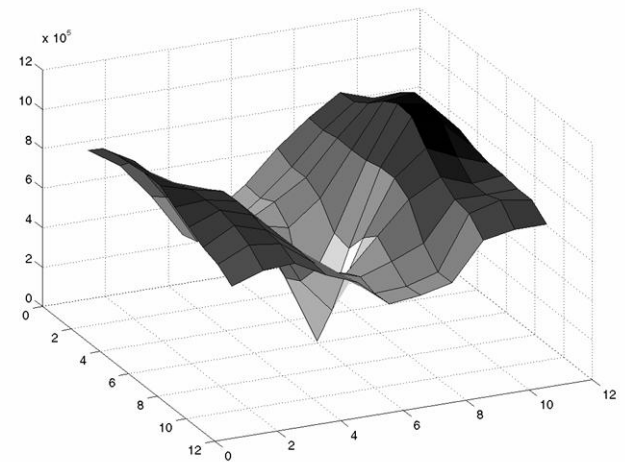
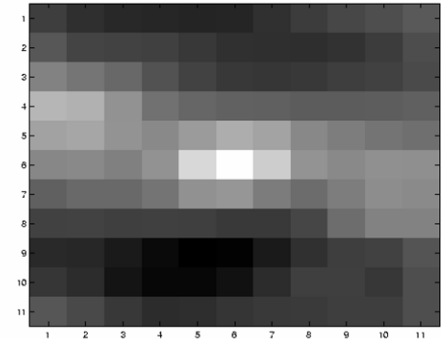
Contour rectiligne : 1 seule
grande valeur propre



Fonction d'appariement A

Sources : Steven Seitz UW-Seattle

Rang 2 : point anguleux



Point anguleux : 2 valeurs
propres importantes

Sources : Steven Seitz UW-Seattle

Fonction d'appariement A

Techniques différentielles (2)

HORN & SCHUNCK 1981

Exploitation directe de la contrainte (1) :

$$I(x, y, t) = I(x + \delta x, y + \delta y, t + \delta t)$$

Formule de Taylor

$$= I(x, y, t) + \frac{\partial I}{\partial x} \cdot \delta x + \frac{\partial I}{\partial y} \cdot \delta y + \frac{\partial I}{\partial t} \cdot \delta t + \dots$$

Termes d'ordres supérieures négligés

SOIT :

$$\boxed{\nabla I \cdot v + \frac{\partial I}{\partial t} = 0}$$

avec

$$\left\{ \begin{array}{ll} \nabla I = \left(\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y} \right) & \text{gradient spatial} \\ v = (v_x^t, v_y^t) & \text{inconnues} \\ \frac{\partial I}{\partial t} & \text{gradient temporel} \end{array} \right.$$

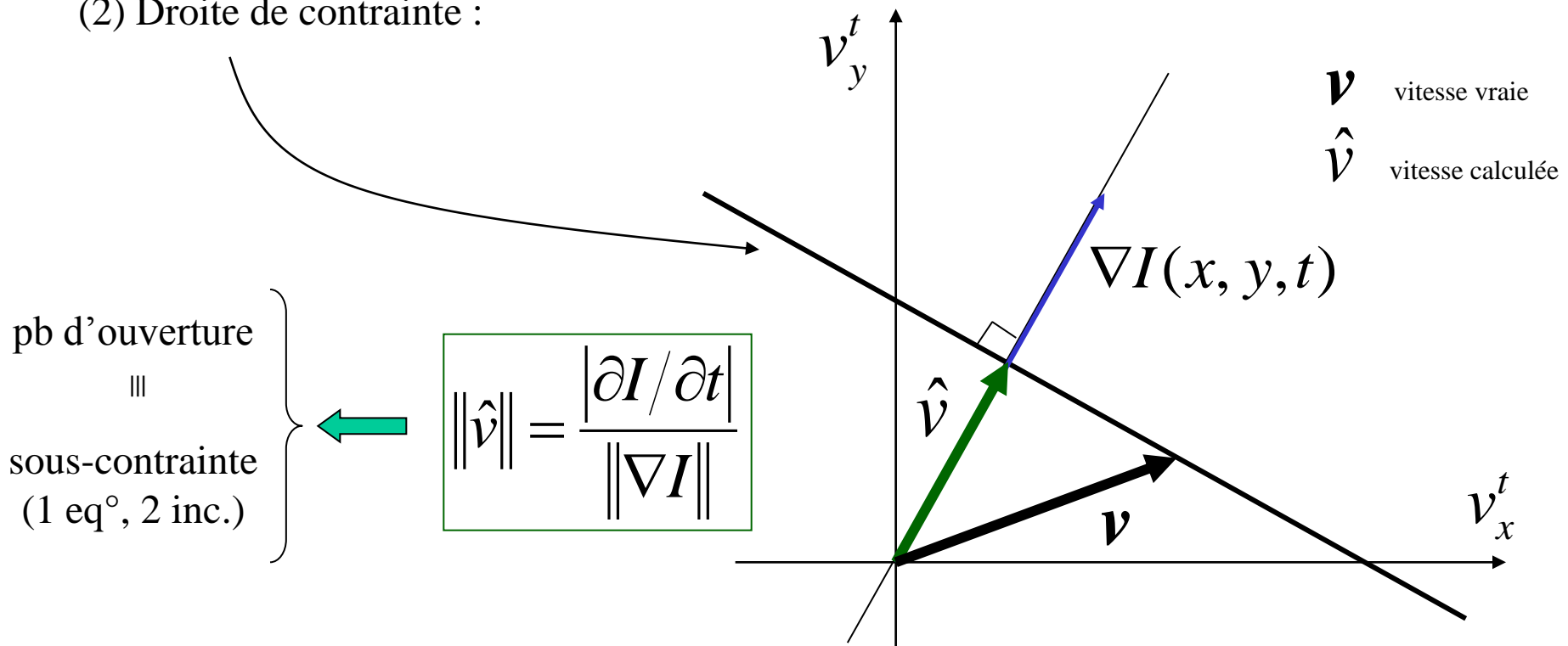
Equation de contrainte du mouvement apparent (ECMA)

ou : équation du flot optique

Interprétation de l'ECMA

(1) En faisant l'hypothèse d'une certaine *régularité* du champ et de *petits déplacements*, les *changements temporels* dans l'image sont *équivalents* (au premier ordre) au produit scalaire des *changements spatiaux* et de la *vitesse apparente*.

(2) Droite de contrainte :



Résolution de l'ECMA

HORN & SCHUNCK 1981

Résolution de l'ECMA par *ajout d'une contrainte de régularité*.

⇔ *Régularisation* du pb mal posé par hypothèse de champ *lisse* de déplacement.

→ Minimisation d'une fonction de coût :

$$C_{(x,y)}^t(v_x^t, v_y^t) = \underbrace{\left(\nabla I \cdot v + \frac{\partial I}{\partial t} \right)^2}_{\text{ECMA}} + \lambda \underbrace{\left[\left(\frac{\partial v_x^t}{\partial x} \right)^2 + \left(\frac{\partial v_x^t}{\partial y} \right)^2 + \left(\frac{\partial v_y^t}{\partial x} \right)^2 + \left(\frac{\partial v_y^t}{\partial y} \right)^2 \right]}_{\text{REGULARISATION}}$$

ECMA

REGULARISATION

facteur de
pondération

Résolution de l'ECMA

HORN & SCHUNCK 1981

1. Minimisation d'une fonctionnelle quadratique :

$$C_{(x,y)}(u,v) = (I_x u + I_y v + I_t)^2 + \lambda(u_x^2 + u_y^2 + v_x^2 + v_y^2)$$

$$\text{avec : } \begin{cases} (u,v) \text{ composantes du champ (inconnues) à calculer} \\ \{I_x, I_y, I_t\} \text{ dérivées partielles de l'image} \\ \{u_x, u_y, v_x, v_y\} \text{ dérivées partielles des composantes du champ} \end{cases}$$

—————→ Annulation des dérivées premières $\partial/\partial u(\dots) = 0$; $\partial/\partial v(\dots) = 0$

Equations d'Euler-Lagrange de minimisation de C :

$$\begin{cases} 2(I_x u + I_y v + I_t)I_x + 2\lambda \left[\left(\frac{\partial}{\partial u} \frac{\partial u}{\partial x} \right) \frac{\partial u}{\partial x} + \left(\frac{\partial}{\partial u} \frac{\partial u}{\partial y} \right) \frac{\partial u}{\partial y} \right] = 0 \\ 2(I_x u + I_y v + I_t)I_y + 2\lambda \left[\left(\frac{\partial}{\partial v} \frac{\partial v}{\partial x} \right) \frac{\partial v}{\partial x} + \left(\frac{\partial}{\partial v} \frac{\partial v}{\partial y} \right) \frac{\partial v}{\partial y} \right] = 0 \end{cases}$$

$$\text{soit : } \begin{cases} (I_x u + I_y v + I_t)I_x + \lambda \Delta u = 0 \\ (I_x u + I_y v + I_t)I_y + \lambda \Delta v = 0 \end{cases} \quad \text{avec : } \begin{cases} \Delta u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \\ \Delta v = \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \end{cases} \quad \text{laplaciens de } u \text{ et } v$$

Résolution de l'ECMA

2. Approximation du Laplacien : $\nabla^2 f = f - \tilde{f}$ \tilde{f} = moyenne de f dans un certain voisinage

soit :
$$\begin{cases} (v - \tilde{v})(I_x^2 + I_y^2 + \lambda) = -I_y(I_x \tilde{u} + I_y \tilde{v} + I_t) \\ (u - \tilde{u})(I_x^2 + I_y^2 + \lambda) = -I_x(I_x \tilde{u} + I_y \tilde{v} + I_t) \end{cases}$$

3. Schéma itératif de résolution :

Méthode de Gauss-Seidel :
$$\begin{cases} u^k = \tilde{u}^{k-1} - I_x(I_x \tilde{u}^{k-1} + I_y \tilde{v}^{k-1} + I_t) \cdot (I_x^2 + I_y^2 + \lambda)^{-1} \\ v^k = \tilde{v}^{k-1} - I_y(I_x \tilde{u}^{k-1} + I_y \tilde{v}^{k-1} + I_t) \cdot (I_x^2 + I_y^2 + \lambda)^{-1} \end{cases}$$

Soit, en reprenant les notations originales :

ALGORITHME DE HORN & SCHUNCK

avec :

$$\begin{cases} N = \frac{\partial I}{\partial x} \tilde{v}_x^{k-1} + \frac{\partial I}{\partial y} \tilde{v}_y^{k-1} + \frac{\partial I}{\partial t} \\ D = \lambda + \left(\frac{\partial I}{\partial x} \right)^2 + \left(\frac{\partial I}{\partial y} \right)^2 \end{cases}$$

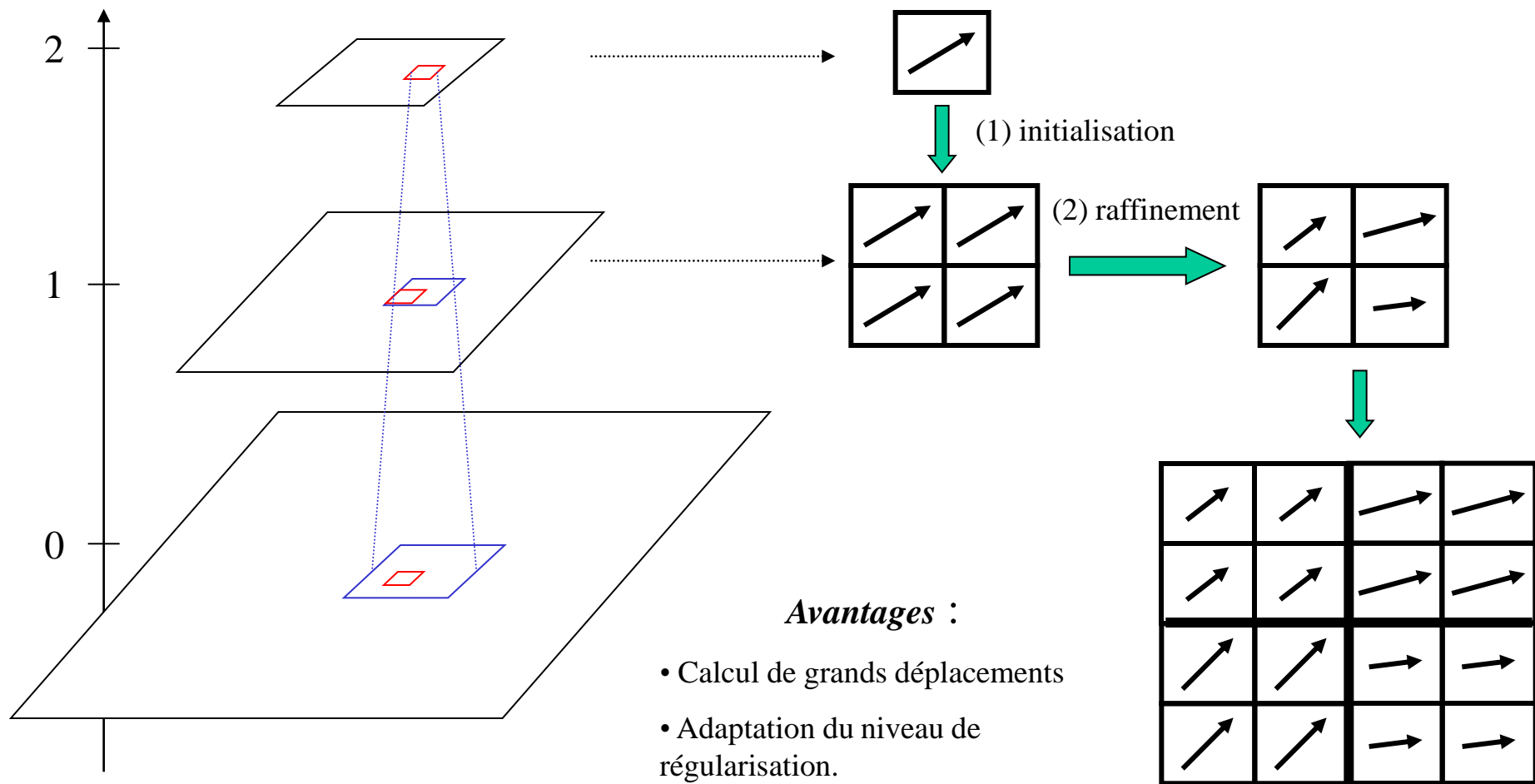
• Initialisation : $v_x^0 = 0$ $v_y^0 = 0$

• Répéter jusqu'à convergence :

$$\begin{cases} v_x^k = \tilde{v}_x^{k-1} - \frac{\partial I}{\partial x} \frac{N}{D} \\ v_y^k = \tilde{v}_y^{k-1} - \frac{\partial I}{\partial y} \frac{N}{D} \end{cases}$$

Estimation multi-échelles

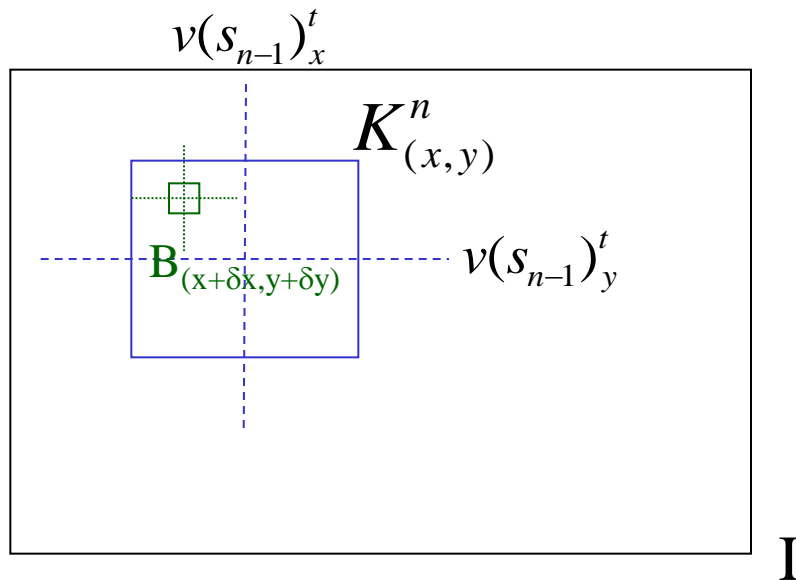
Niveau



Principe du calcul multi-échelles

ALGORITHME PAR CORRELATION

$$(v(s_n)_x, v(s_n)_y) = \arg \max_{(\delta x, \delta y) \in K^n} C_{(x,y)}^t(\delta x, \delta y)$$



avec : $K^n_{(x,y)} = \left[v(s_{n-1})^t_x \pm \Delta_1 \right] \times \left[v(s_{n-1})^t_y \pm \Delta_2 \right]$

ALGORITHME DIFFERENTIEL

- Initialisation : $v(s_0)_x^0 = 0$ $v(s_0)_y^0 = 0$
- Répéter jusqu'à convergence :

$$\begin{cases} v(s_0)_x^k = \tilde{v}(s_0)_x^{k-1} - \frac{\partial I}{\partial x} \frac{N}{D} \\ v(s_0)_y^k = \tilde{v}(s_0)_y^{k-1} - \frac{\partial I}{\partial y} \frac{N}{D} \end{cases}$$



- Init. : $v(s_1)_x^0 = v(s_0)_x^\infty$ $v(s_1)_y^0 = v(s_0)_y^\infty$
- Répéter jusqu'à convergence :

$$\begin{cases} v(s_1)_x^k = \tilde{v}(s_1)_x^{k-1} - \frac{\partial I}{\partial x} \frac{N}{D} \\ v(s_1)_y^k = \tilde{v}(s_1)_y^{k-1} - \frac{\partial I}{\partial y} \frac{N}{D} \end{cases}$$



.../...

Technique du plus proche voisin

La technique du plus proche voisin projeté consiste à :

- (1) Projeter les pixels de l'espace-temps image dans un espace de caractéristiques (*feature space*), dans lequel la distance soit directement reliée à la similarité.
- (2) Trouver les pixels les plus ressemblants d'une image à l'autre en recherchant le plus proche voisin dans l'espace de caractéristiques.

Quel espace de caractéristiques ? → Dérivées partielles (*Local Jet*) multi-échelles.

Mesure de similarité par SSD :

$$d_I(\mathbf{p}, \mathbf{q}) = \sum_{\mathbf{i} \in B} (I(\mathbf{p} + \mathbf{i}) - I(\mathbf{q} + \mathbf{i}))^2$$

Formule de Taylor à l'ordre r :

$$I(\mathbf{p} + \mathbf{c}) = \sum_{k=0}^r \sum_{i=0}^k C_k^i \frac{\partial^k I}{\partial^{k-i} x \partial^i y} \mathbf{c}_x^{k-i} \mathbf{c}_y^i + o(\|\mathbf{c}\|^r)$$

$$d_I(\mathbf{p}, \mathbf{q}) \approx \sum_{\mathbf{i} \in B} \alpha_{ij} (I_{ij}(\mathbf{p}) - I_{ij}(\mathbf{q}))^2$$

Mesure de similarité par distance euclidienne dans l'espace des dérivées partielles (*local jet*)

$$I_{ij} = \frac{\partial^{i+j} I}{\partial^i x \partial^j y}$$

Projection dans l'espace des local jets

Les vecteurs de caractéristiques sont les composantes normalisées du local jet :

$$\mathbf{p} \mapsto \hat{\mathbf{p}} = \left(a_{ij}^{\sigma} I_{ij}^{\sigma}(\mathbf{p}) \right)_{i+j \leq r}$$

avec :

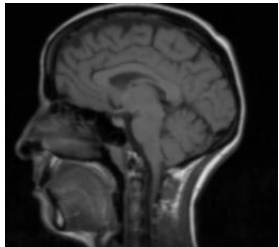
$$I_{ij}^{\sigma} = I * \frac{\partial^{i+j} G_{\sigma}}{\partial^i x \partial^j y}$$

Gaussienne bidimensionnelle de variance σ^2

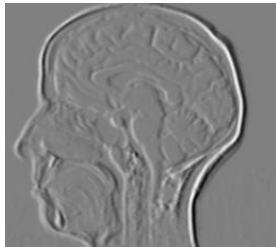
et :

$$a_{ij}^{\sigma} = \frac{\sigma^{i+j}}{i+j+1}$$

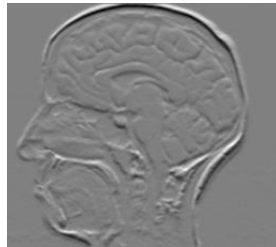
Normalisation scale-space
Nombre de dérivées $(i+j)^{\text{èmes}}$



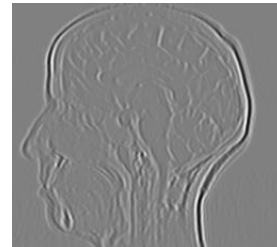
$I_{00}^{1.0}$



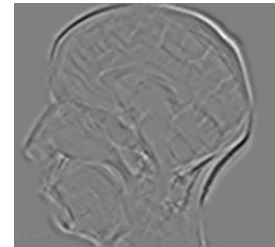
$I_{10}^{1.0}$



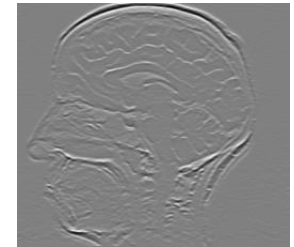
$I_{01}^{1.0}$



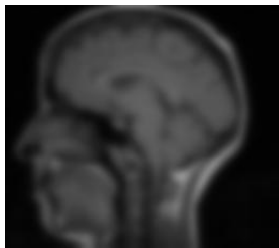
$I_{20}^{1.0}$



$I_{11}^{1.0}$



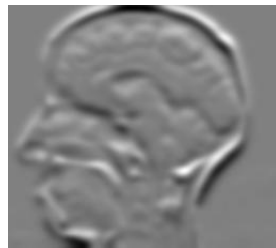
$I_{02}^{1.0}$



$I_{00}^{4.0}$



$I_{10}^{4.0}$



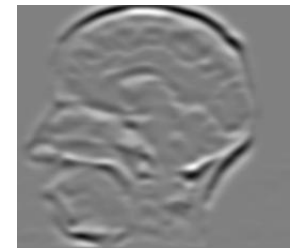
$I_{01}^{4.0}$



$I_{20}^{4.0}$

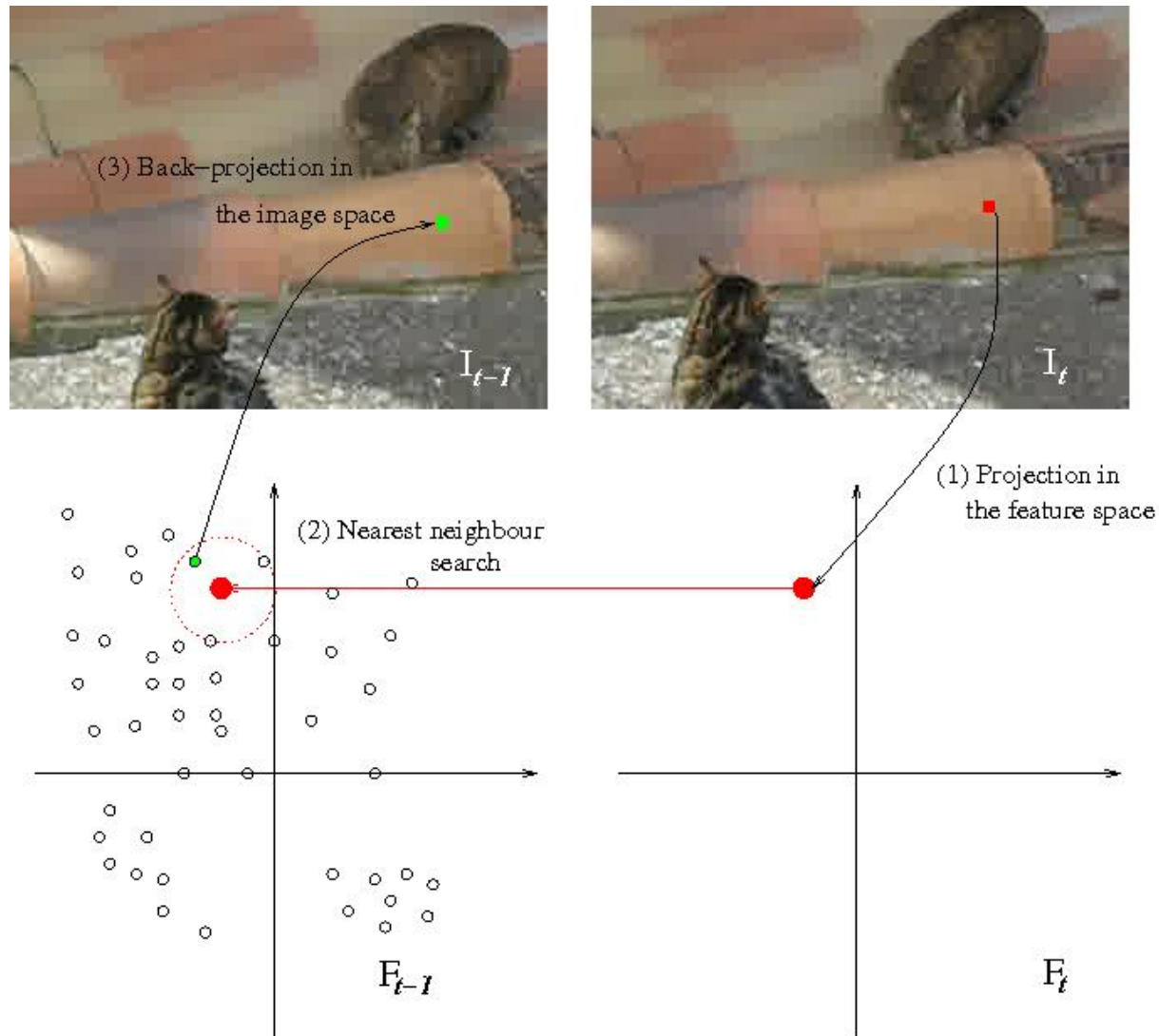


$I_{11}^{4.0}$

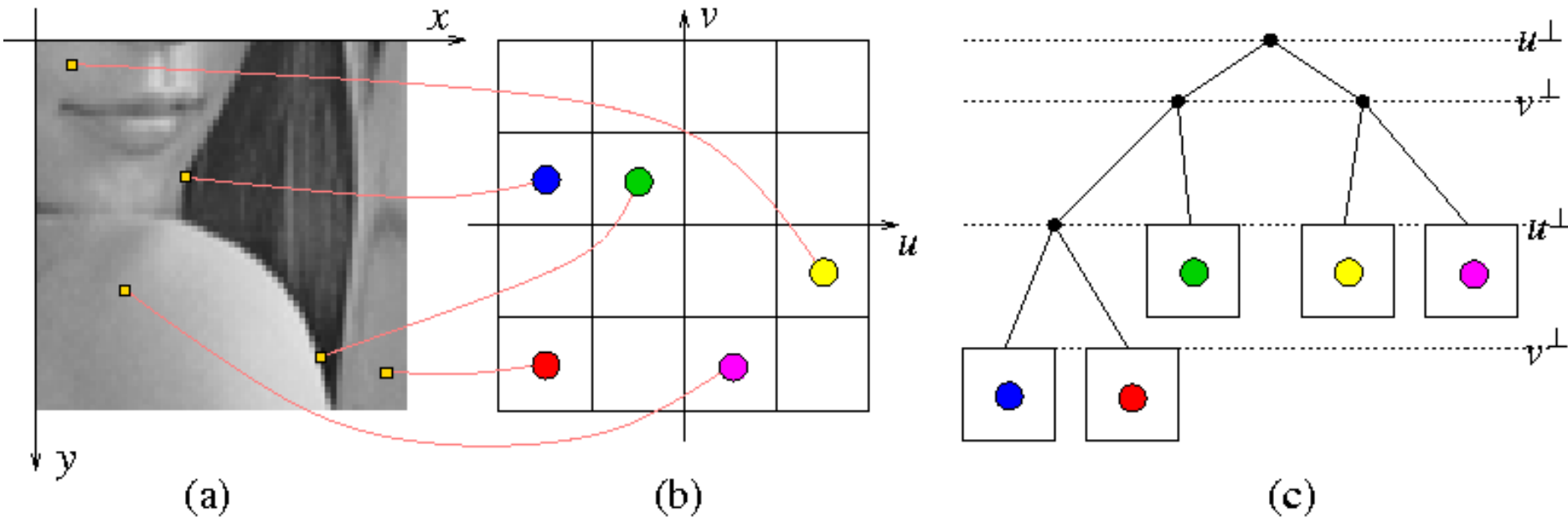


$I_{02}^{4.0}$

Recherche du plus proche voisin

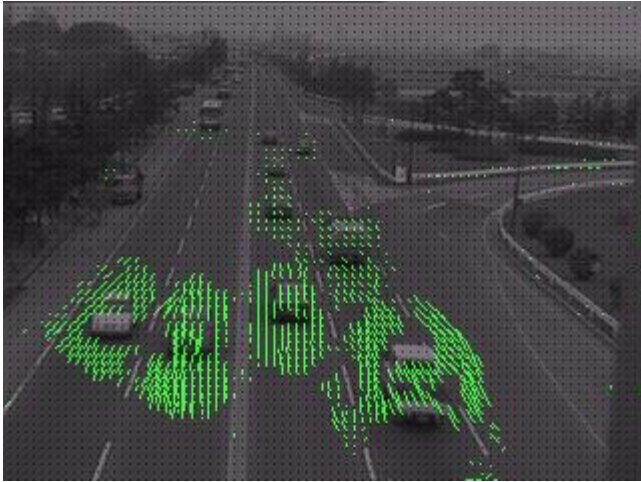


Représentation et Recherche du PPV



Il faut construire une représentation adaptée de l'espace des caractéristiques (EC), pour une recherche efficace du plus proche voisin (PPV). Le tableau multidimensionnel (a) est souvent optimal en temps mais très coûteux en mémoire quand la dimension de l'EC est grande. On utilise plutôt une représentation de type kd-tree (b), optimale en espace, et avec un coût de recherche moyen du PPV en $O(\log(N))$ (où N est le nombre de vecteurs de caractéristiques).

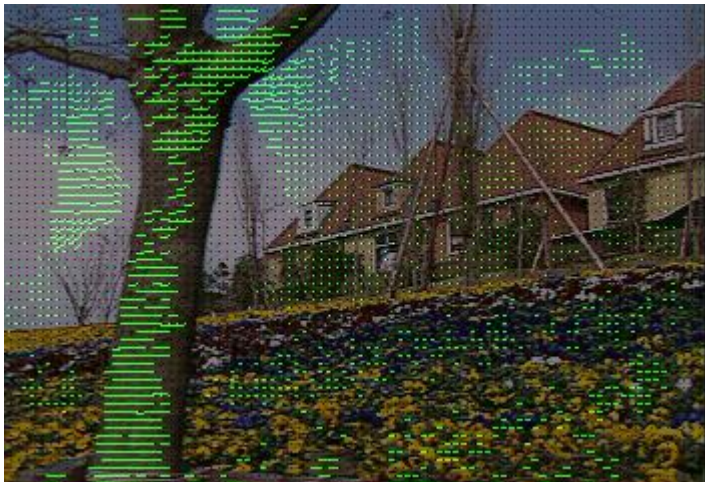
Technique du plus proche voisin



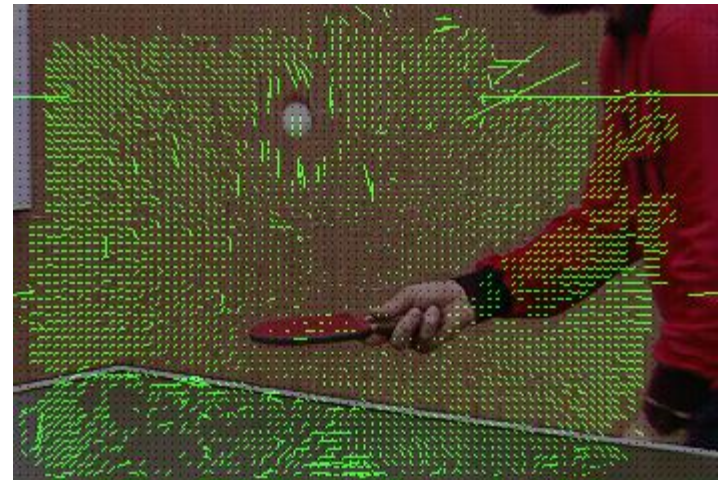
Caméra fixe – objets en mouvement radial



Zoom avant + Mouvement fluide des nuages



Travelling latéral



Zoom avant + balle mobile

Conclusion Chapitre 1

☐ EXPLOITATION DU FLOT OPTIQUE

- ☐ Mouvement dominant
- ☐ Profondeur
- ☐ Temps avant collision

☐ METHODES PAR APPARIEMENT

☐ METHODES DIFFERENTIELLES

- ☐ Lucas et Kanade
- ☐ Horn et Schunck

☐ METHODES MULTI-ECHELLES

☐ METHODES PAR PLUS PROCHE VOISIN PROJETE

Bibliographie Chapitre 1

- **B.D. Lucas & T. Kanade 1981** « *An iterative image registration technique with an application to stereo vision* » *International Journal of Computer Vision and Artificial Intelligence* 674-679
- **B.K.P Horn & B. Schunck 1981** « *Determining Optical Flow* » *Artificial Intelligence* 23 185-203
- **D.H. Ballard & C.M Brown 1982** « *Computer Vision* » Prentice Hall (Ch. 3, Ch. 7)
- **J.M. Jolion & A. Rosenfeld 1994** « *A pyramid framework for early vision* » Kluwer Academic Publishers Dordrecht, NL
- **R. Jain, R. Kasturi & B. Schunck 1995** « *Machine Vision* » McGraw-Hill Inc. (Ch.14)
- **A. Manzanera 2010** « *Image representation and processing through multiscale local jet* » Research Report - http://www.ensta.fr/~manzaner/Publis/LJFeat_AM2010.pdf

PLAN DU CHAPITRE

- **Recalage : méthodes fréquentielles**
- **Poursuite 1 : Harris et Invariants de Hilbert**
- **Poursuite 2 : SIFT**

Introduction

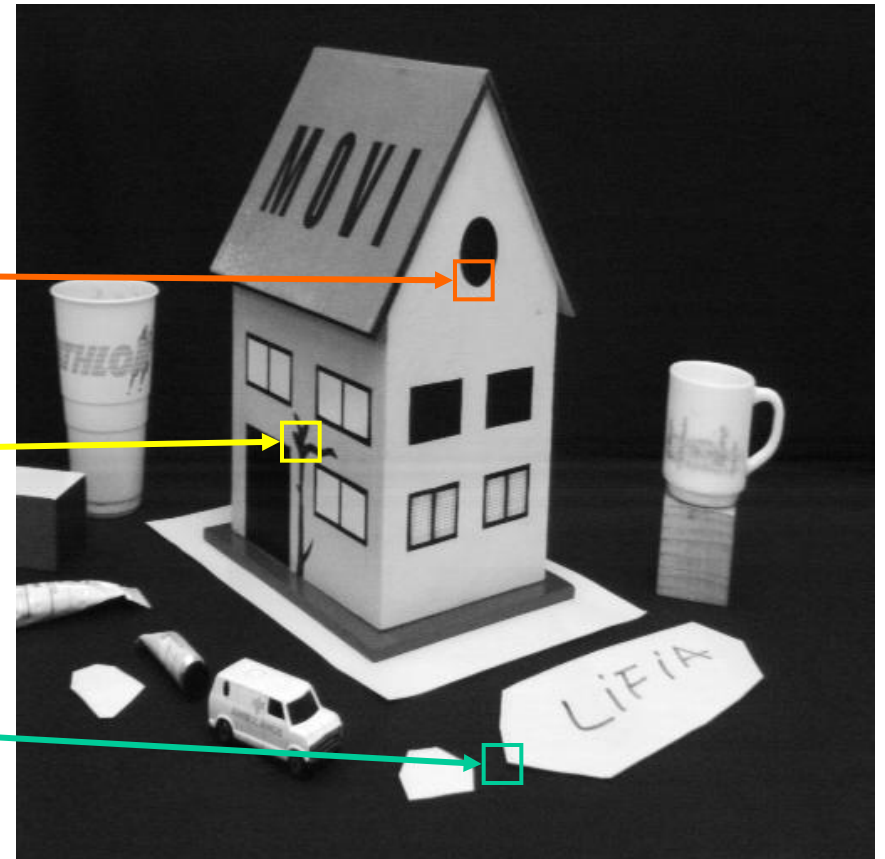
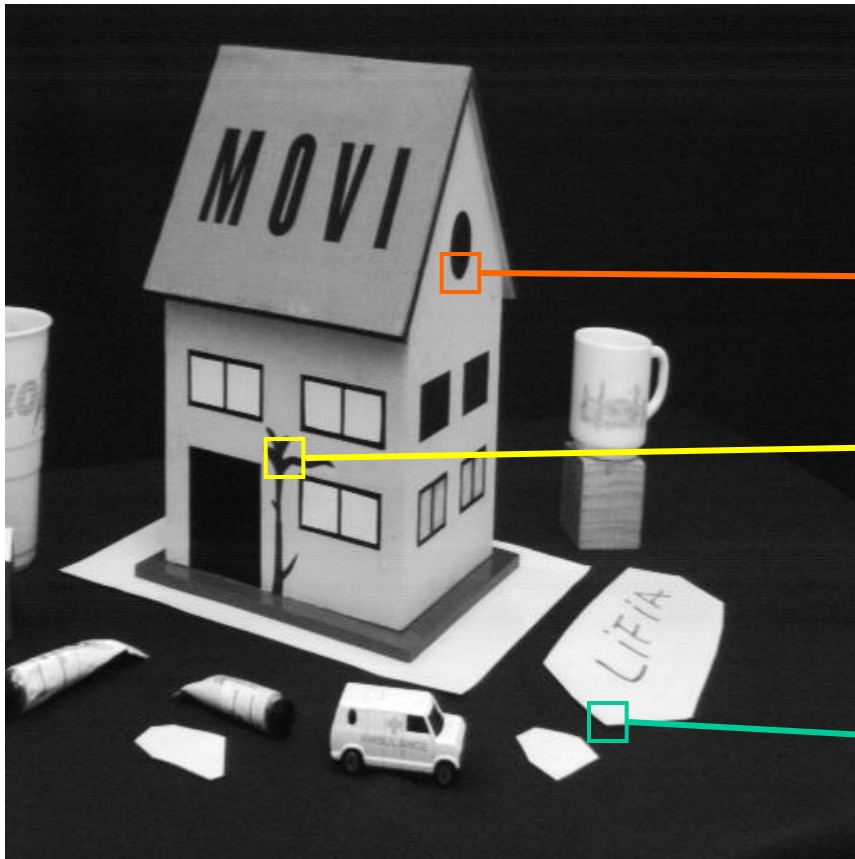


Lorsque l'amplitude du déplacement est grande, l'estimation directe par les méthodes d'appariement ou les méthodes différentielles est vouée à l'échec, pour 2 raisons majeures :

(1) Le déplacement apparent peut être supérieur aux périodes spatiales présentes dans les images (aliasing temporel, voir exemple ci-contre)

Introduction

(2) La scène est soumise à des déformations géométriques complexes qui invalident l'hypothèse de translation locale (voir exemple ci-dessous)



Introduction

Par conséquent :

- Il est primordial de disposer de descripteurs qui soit le plus *invariants* possible aux *transformations géométriques* : rotation, homothétie, transformation affine. Cela passe forcément par un calcul multi-échelle.
- Les descripteurs fournis auront un caractère beaucoup *moins local*, voire global, et on ne disposera plus d'un champ dense comme dans les méthodes d'estimation, mais au mieux d'un *champ épars*.

Ce qu'on verra dans ce chapitre :

TRANSFORMATION GLOBALE :

- Un seul paramètre de déplacement pour toute l'image (translation, rotation, homothétie) : *Méthodes fréquentielles*.

MOUVEMENT QUELCONQUE :

- Quels sont les points utilisés pour le calcul ? *Points de Harris, Maxima locaux dans l'espace d'échelle,...*
- Quels descripteurs utiliser pour l'appariement ? *Invariants différentiels, histogrammes d'orientation,...*

Applications : Reconnaissance d'objets, Indexation d'images, localisation de robots,...

Techniques fréquentielles

Les techniques fréquentielles d'estimation du mouvement entre deux images sont fondées sur l'équivalence translation/déphasage de la transformée de Fourier :

Rappel : l'expression d'une image dans le domaine fréquentiel consiste à décomposer la fonction bidimensionnelle en sommes de sinusoides complexes :

$$I(x, y) = \frac{1}{wh} \sum_{u=0}^{w-1} \sum_{v=0}^{h-1} F(u, v) e^{2i\pi(ux+vy)/wh}$$

*Transformée de Fourier
discrète inverse*



Les coefficients des différentes sinusoides sont calculés par la transformée de Fourier :

$$F(u, v) = \sum_{x=0}^{w-1} \sum_{y=0}^{h-1} I(x, y) e^{-2i\pi(ux+vy)/wh}$$

*Transformée de Fourier
discrète directe*

Notation (module, phase) : $F(u, v) = \|F(u, v)\| e^{i\varphi_F(u, v)}$

La propriété de translation/déphasage dit que si F est la transformée de Fourier de I :

Alors la TF de I translatée de $(-\delta x, -\delta y)$, est G , avec :

$$\begin{array}{ccc} I(x, y) & \xrightarrow{\text{TF}} & F(u, v) \\ I(x + \delta x, y + \delta y) & \xrightarrow{\text{TF}} & G(u, v) = F(u, v) e^{2i\pi(u\delta x + v\delta y)/wh} \end{array}$$

Soit : $\|G(u, v)\| = \|F(u, v)\|$ et : $\varphi_G(u, v) = \varphi_F(u, v) + 2\pi(u\delta x + v\delta y)/wh$

Le déphasage entre F et G vaut donc : $\Delta\phi(u, v) = 2\pi(u\delta x + v\delta y)/wh$

Il suffit donc en théorie de considérer ce déphasage pour 2 couples (u, v) pour calculer $(\delta x, \delta y)$, mais cette technique est sensible au bruit et aux changements d'illumination qui induisent des variations dans les basses fréquences.

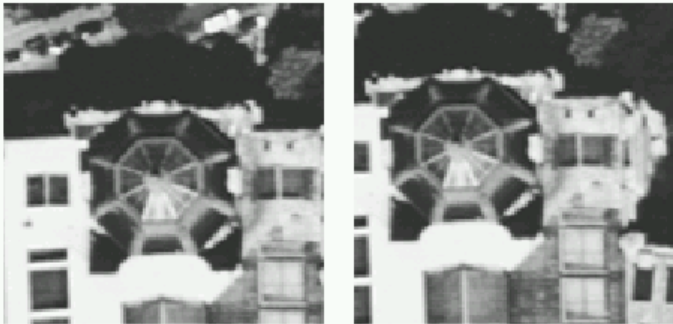
On utilise plutôt la technique de corrélation de phase.

Corrélation de phase

La technique de corrélation de phase exploite une conséquence directe de la propriété de translation/déphasage. Si F est la TF de I et G la TF de I translatée de $(-\delta x, -\delta y)$, alors le déphasage entre F et G est égal à leur spectre de puissance croisé normalisé (SPCN), i.e. :

$$\frac{F^*(u, v)G(u, v)}{\|F^*(u, v)G(u, v)\|} = e^{2i\pi(u\delta x + v\delta y)/wh}$$

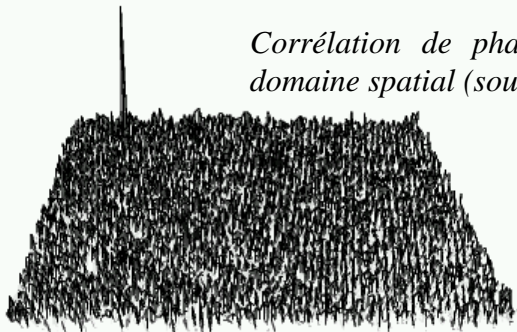
La TF inverse du SPCN est donc égale à la fonction de Dirac du vecteur de translation : $\delta_{(\delta x, \delta y)}(x, y)$



(a)

(b)

Corrélation de phase dans le domaine spatial (source INRIA)



(c)

La technique de corrélation de phase consiste donc à :

1. Calculer les TF de $I(x, y, t)$ et $I(x, y, t+1)$, soit F_1 et F_2
2. Calculer χ le SPCN de F_1 et F_2
3. Calculer D la TF inverse de χ
4. Rechercher le maximum de D

Avantages et inconvénients

- + Robuste car toutes les fréquences contribuent au calcul
- + Relativement rapide grâce au calcul de la FFT
- En pratique limité à un déplacement global sur toute l'image

Invariant de Fourier-Mellin

L'utilisation de la transformée de *Fourier-Mellin* permet de calculer les paramètres d'une similitude (*rotation et homothétie*) comme un *vecteur de translation* de manière analogue au cas précédent, grâce à une représentation log-polaire de l'espace des fréquences $(u, v) \rightarrow (\theta, \log \rho)$:

Soit g l'image transformée de f par une rotation d'angle α , une homothétie de rapport σ , et une translation de vecteur (x_0, y_0) :

$$g(x, y) = f(\sigma(\cos \alpha x + \sin \alpha y) - x_0, \sigma(-\sin \alpha x + \cos \alpha y) - y_0)$$

Les amplitudes des transformées de Fourier de f et g sont liées par la relation suivante :

$$\|G(u, v)\| = \frac{1}{\sigma^2} \|F(\frac{1}{\sigma}(u \cos \alpha + v \sin \alpha), \frac{1}{\sigma}(-u \sin \alpha + v \cos \alpha))\|$$

donc l'amplitude :

- ne dépend pas de la translation (x_0, y_0) .
- subit une rotation d'angle α .
- subit une modification d'échelle d'un facteur $1/\sigma$.

En passant les fréquences en coordonnées polaires :

$$F_p(\theta, \rho) = \|F(\rho \cos \theta, \rho \sin \theta)\|; 0 \leq \theta \leq 2\pi, 0 \leq \rho < \infty$$

$$G_p(\theta, \rho) = \|G(\rho \cos \theta, \rho \sin \theta)\|; 0 \leq \theta \leq 2\pi, 0 \leq \rho < \infty$$

Enfin, en passant la coordonnée radiale au logarithme :

$$r = \log \rho$$

$$s = \log \sigma$$

$$F_{lp}(\theta, r) = F_p(\theta, \rho)$$

$$G_{lp}(\theta, r) = G_p(\theta, \rho)$$

on obtient :

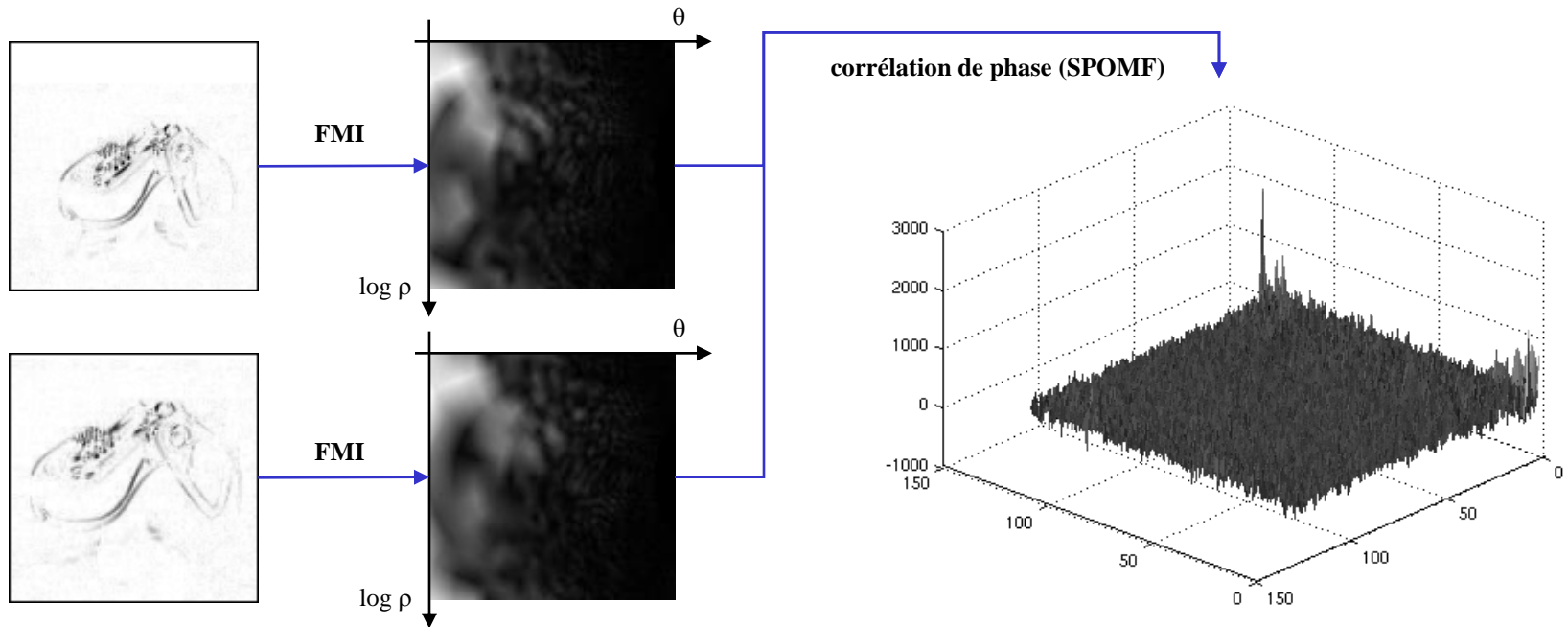
$$G_p(\theta, \rho) = \frac{1}{\sigma^2} F_p\left(\theta - \alpha, \frac{\rho}{\sigma}\right)$$

on obtient :

$$G_{lp}(\theta, r) = \frac{1}{\sigma^2} F_{lp}(\theta - \alpha, r - s)$$

Donc une similitude dans l'espace image se traduit par une *translation* dans l'espace des fréquences *log-polaires*.

Invariant de Fourier-Mellin



Un exemple d'utilisation de la transformée de Fourier-Mellin : calcul de la position de la tête des Robots Aibo dans l'image par corrélation de phase des invariants de Fourier-Mellin. (FMI-SPOMF : Fourier-Mellin Invariant Symmetric Phase Only Matched Filtering) : **J.C. Baillie et M. Nottale** 2004.



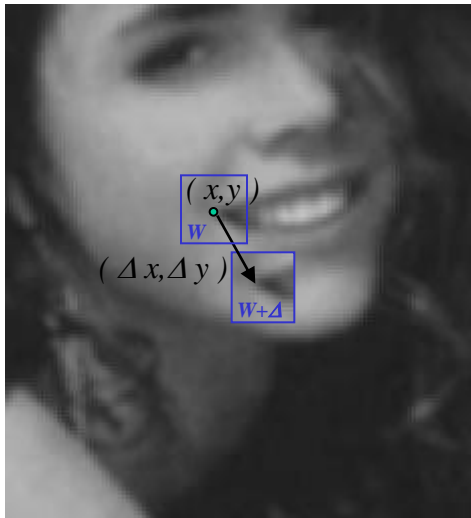
L'information de phase de l'image originale est perdue dans la FMI. Le FMI-SPOMF revient à chercher la meilleure (rotation, homothétie) qui mette en correspondance 2 spectres d'amplitude. *On ne retrouve donc pas les paramètres de translation entre les deux images, et de plus l'information de forme portée par la phase n'existe plus.*

Pour compléter cette transformation, on peut appliquer une corrélation de phase classique sur le couple d'image de départ, après avoir appliqué sur l'une des images la transformation (rotation, homothétie) fournie par le FMI-SPOMF.

Notons enfin, que comme pour la corrélation de phase, cette méthode est utilisée en pratique pour estimer des transformations *globales*, car elle utilise la contribution de tout le spectre (ou au moins une large partie), ce qui implique une étendue spatiale importante des pixels utilisés pour l'estimation de chaque transformation.

Détection de points anguleux

Les points anguleux (ou points d'intérêt, points saillants,...) sont des points « qui contiennent beaucoup d'information » relativement à l'image. Ce sont des points aux voisinages desquels l'image *varie significativement dans plusieurs directions*.



Une mesure des variations locales de l'image I au point (x, y) associée à un déplacement $(\Delta x, \Delta y)$ est fournie par la *fonction d'autocorrélation* :

$$\chi(x, y) = \sum_{(x_k, y_k) \in W} (I(x_k, y_k) - I(x_k + \Delta x, y_k + \Delta y))^2$$

Où W est une fenêtre centrée au point (x, y) .

Or, en utilisant une approximation du premier ordre :

$$I(x_k + \Delta x, y_k + \Delta y) \approx I(x_k, y_k) + \left(\frac{\partial I}{\partial x}(x_k, y_k) \quad \frac{\partial I}{\partial y}(x_k, y_k) \right) \cdot \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix}$$

Et donc :

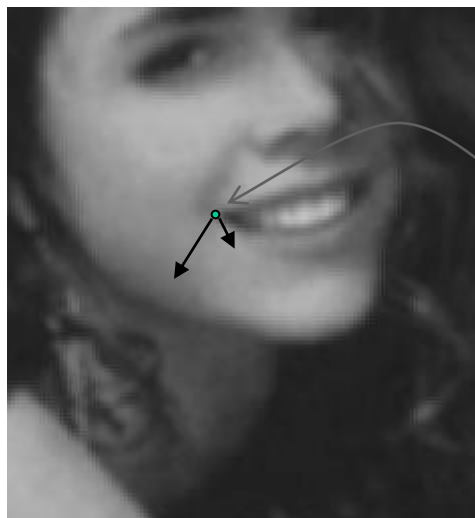
$$\chi(x, y) = \sum_{(x_k, y_k) \in W} \left(\left(\frac{\partial I}{\partial x}(x_k, y_k) \quad \frac{\partial I}{\partial y}(x_k, y_k) \right) \cdot \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} \right)^2 = \begin{pmatrix} \Delta x & \Delta y \end{pmatrix} \underbrace{\begin{pmatrix} \sum_{(x_k, y_k) \in W} \left(\frac{\partial I}{\partial x}(x_k, y_k) \right)^2 & \sum_{(x_k, y_k) \in W} \frac{\partial I}{\partial x}(x_k, y_k) \cdot \frac{\partial I}{\partial y}(x_k, y_k) \\ \sum_{(x_k, y_k) \in W} \frac{\partial I}{\partial x}(x_k, y_k) \cdot \frac{\partial I}{\partial y}(x_k, y_k) & \sum_{(x_k, y_k) \in W} \left(\frac{\partial I}{\partial y}(x_k, y_k) \right)^2 \end{pmatrix}}_{\Xi(x, y)} \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix}$$

Matrice d'autocorrélation de l'image I en (x, y)

Détection de points anguleux

$$\Xi(x, y) = \begin{pmatrix} \sum_{(x_k, y_k) \in W} \left(\frac{\partial I}{\partial x}(x_k, y_k) \right)^2 & \sum_{(x_k, y_k) \in W} \frac{\partial I}{\partial x}(x_k, y_k) \cdot \frac{\partial I}{\partial y}(x_k, y_k) \\ \sum_{(x_k, y_k) \in W} \frac{\partial I}{\partial x}(x_k, y_k) \cdot \frac{\partial I}{\partial y}(x_k, y_k) & \sum_{(x_k, y_k) \in W} \left(\frac{\partial I}{\partial y}(x_k, y_k) \right)^2 \end{pmatrix}$$

La matrice d'autocorrélation Ξ représente la variation locale de l'image I en (x, y) . (x, y) sera considéré comme un point anguleux de I si pour tous les déplacements $(\Delta x, \Delta y)$, la quantité $(\Delta x, \Delta y) \cdot \Xi(x, y) \cdot (\Delta x, \Delta y)^t$ est grande.



Les points anguleux sont les points (x, y) pour lesquels la matrice d'autocorrélation $\Xi(x, y)$ a *deux valeurs propres grandes*.

Cela correspond aux points pour lesquels il existe localement une base de vecteurs propres de Ξ décrivant des variations locales importantes de l'image.

Le détecteur de Harris calcule une fonction d'intérêt $\Theta(x, y)$:

$$\Theta(x, y) = \det \Xi - \alpha \text{trace} \Xi^2$$

Le premier terme correspond au produit des valeurs propres, le second terme pénalise les points de contours avec une seule forte valeur propre.

Les points d'intérêt correspondent aux maxima locaux de la fonction Θ qui sont au delà d'un certain seuil (typiquement 1% de la valeur max de Θ).

[Harris 88]

Utilisation des invariants différentiels

Objectif : représenter les points d'intérêt par des *indices* qui soient *invariants* par *rotation* et par *changement d'échelle*.

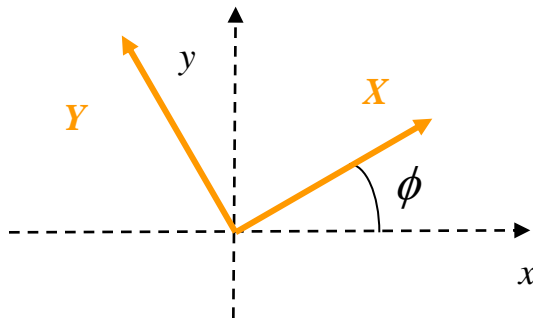
Le principe utilisé ici est basé sur l'utilisation des dérivées spatiales multi-échelle :

Le « jet local » de I : $L_{ij}^\sigma = I * G_{ij}^\sigma$ avec : $G_{ij}^\sigma = \frac{\partial^{i+j}}{\partial x^i \partial y^j} G^\sigma$ et : $G^\sigma(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)$

On notera : $\{L_{ij}^\sigma; 0 \leq i+j \leq 3\} = \{L, L_x, L_y, L_{xx}, L_{xy}, L_{yy}, L_{xxx}, L_{xxy}, L_{xyy}, L_{yyy}\}$ (dérivées jusqu'au 3e ordre)

L'idée est de *combiner ces dérivées* pour obtenir des *grandeurs invariantes par rotation* :

Par exemple, le laplacien $I_{xx} + I_{yy}$ est invariant par rotation :



$$\begin{cases} x = X \cos \phi + Y \sin \phi \\ y = X \sin \phi - Y \cos \phi \end{cases} \quad \begin{cases} X = x \cos \phi + y \sin \phi \\ Y = -x \sin \phi + y \cos \phi \end{cases}$$

$$\begin{cases} I_X = I_x \cos \phi + I_y \sin \phi \\ I_Y = I_x \sin \phi - I_y \cos \phi \end{cases} \quad \text{et :} \quad \begin{cases} I_{XX} = I_{xx} \cos^2 \phi + 2I_{xy} \cos \phi \sin \phi + I_{yy} \sin^2 \phi \\ I_{YY} = I_{xx} \sin^2 \phi - 2I_{xy} \cos \phi \sin \phi + I_{yy} \cos^2 \phi \end{cases}$$

$$\text{et donc : } I_{XX} + I_{YY} = I_{xx} + I_{yy}$$

Utilisation des invariants différentiels

On peut ainsi construire toute une famille de grandeurs invariantes par rotation : les *invariants différentiels de Hilbert*.

$$\Psi = \begin{pmatrix} L \\ L_i L_i \\ L_i L_{ij} L_j \\ L_{ii} \\ L_{ij} L_{ij} \\ \varepsilon_{ij} (L_{jkl} L_i L_k L_l - L_{jkk} L_i L_l L_l) \\ L_{ijj} L_j L_k L_k - L_{ijk} L_i L_j L_k \\ - \varepsilon_{ij} L_{jkl} L_i L_k L_l \\ L_{ijk} L_i L_j L_k \end{pmatrix}$$

Avec :

$$\begin{aligned} \varepsilon_{xx} &= \varepsilon_{yy} = 0 \\ \varepsilon_{xy} &= -\varepsilon_{yx} = 1 \end{aligned}$$

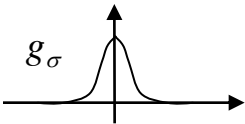
(notations d'Einstein : sommations sur les indices), par ex :

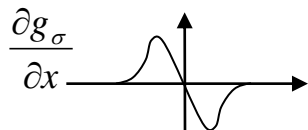
$$\begin{aligned} \Psi_2 &= L_i L_{ij} L_j = L_x L_{xx} L_x + 2L_x L_{xy} L_y + L_y L_{yy} L_y \\ \Psi_7 &= -\varepsilon_{ij} L_{jkl} L_i L_k L_l = L_{xxy} (-L_x L_x L_x + 2L_x L_y L_y) \\ &+ L_{xyy} (-2L_x L_x L_y + L_y L_y L_y) - L_{yyy} L_x L_y L_y + L_{xxx} L_x L_x L_y \end{aligned}$$

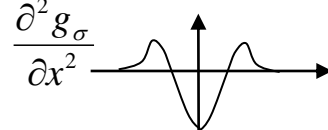
NB : invariance par rotation du noyau gaussien !

Les vecteurs Ψ sont donc calculés pour tous les points d'intérêt à différentes échelles, et appariés en utilisant une distance (e.g. distance euclidienne).

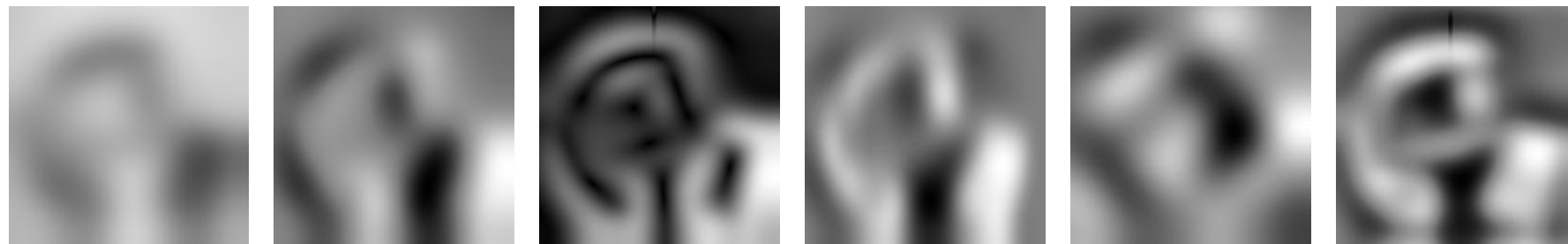
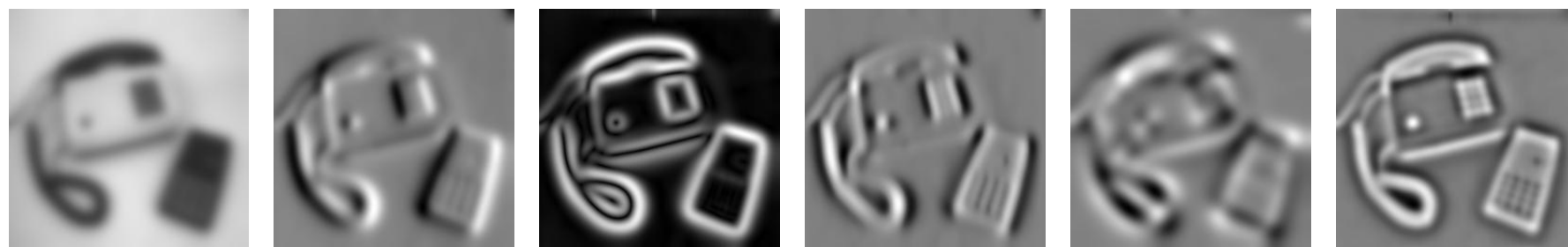
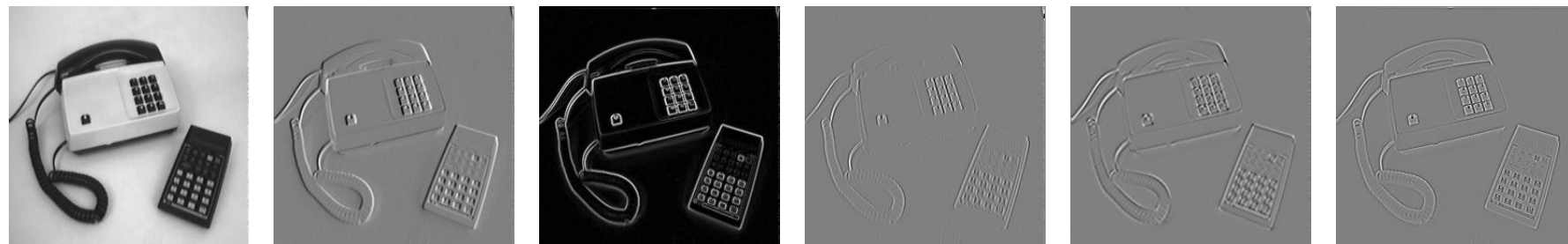
Espace d'échelle gaussien et dérivées

$$g_\sigma(x) = \frac{1}{\pi\sigma} \exp\left(-\frac{x^2}{\sigma}\right)$$


$$\frac{\partial g_\sigma}{\partial x}$$


$$\frac{\partial^2 g_\sigma}{\partial x^2}$$


$$\partial^n (I * g_\sigma) = I * \partial^n g_\sigma$$



I

$$\frac{\partial I}{\partial x}$$

$$\|\nabla I\| = \sqrt{\left(\frac{\partial I}{\partial x}\right)^2 + \left(\frac{\partial I}{\partial y}\right)^2}$$

$$\frac{\partial^2 I}{\partial x^2}$$

$$\frac{\partial^2 I}{\partial x \partial y}$$

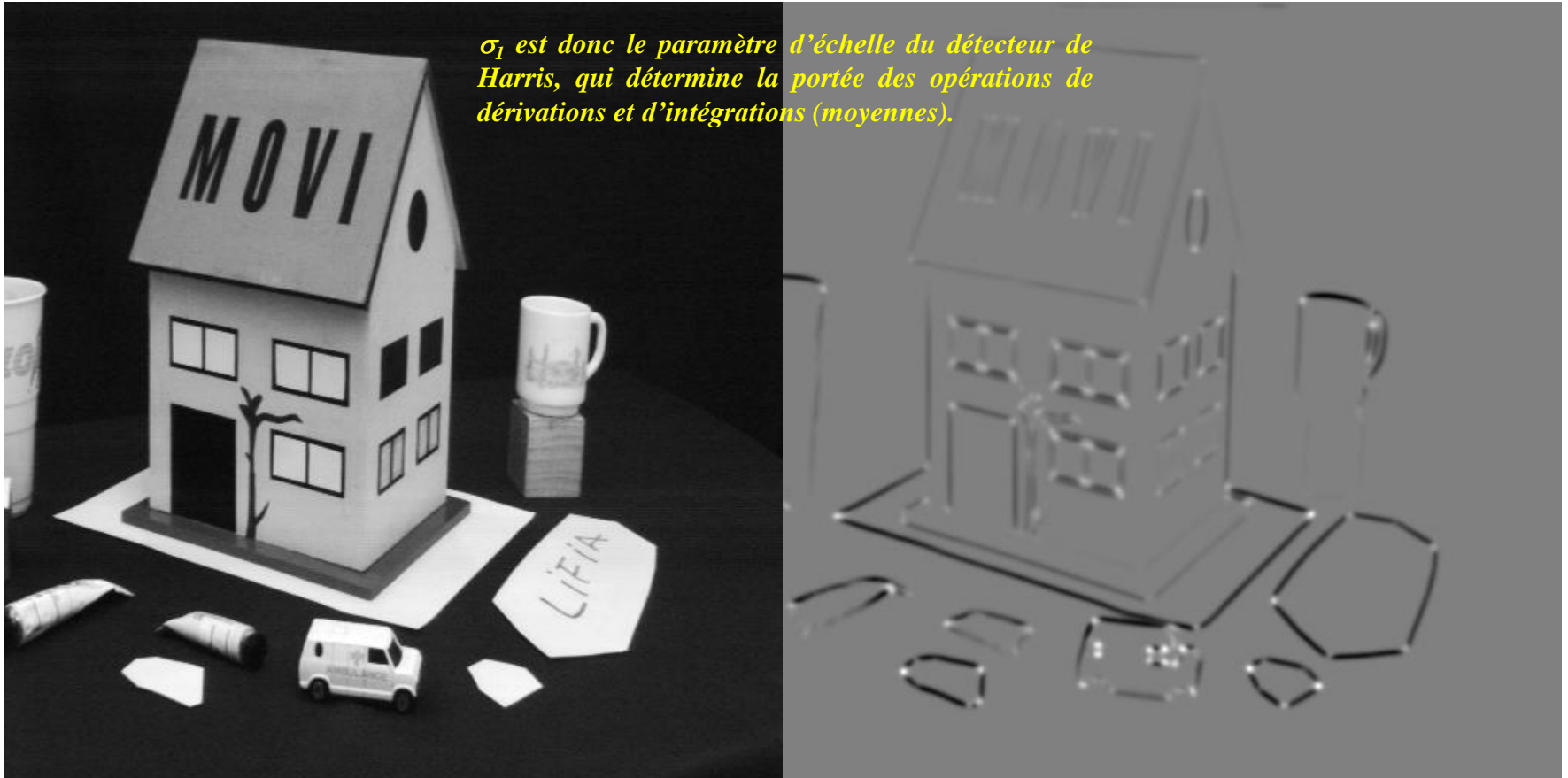
$$\Delta I = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2}$$

échelle

Mise en œuvre du détecteur de Harris

1. On calcule les dérivées premières à partir des dérivées de gaussienne (écart-type σ_1)
2. On calcule les termes de la matrice d'autocorrélation Ξ en calculant une moyenne locale des dérivées sous la forme d'une gaussienne (écart-type σ_2 , typiquement $\sigma_2 = 2 \sigma_1$)
3. On calcule la fonction d'intérêt : $\Theta = \det(\Xi) - \alpha \text{trace}(\Xi)$ (typiquement $\alpha = 0,06$).
4. On calcule les maxima locaux de Θ supérieurs à un certain seuil (typiquement 1% de Θ_{\max}).

σ_1 est donc le paramètre d'échelle du détecteur de Harris, qui détermine la portée des opérations de dérivations et d'intégrations (moyennes).

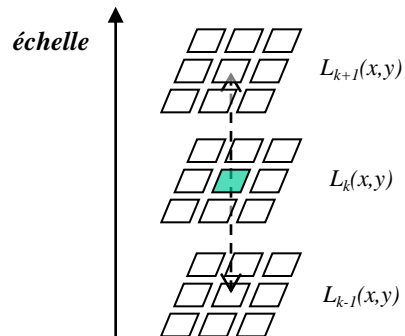
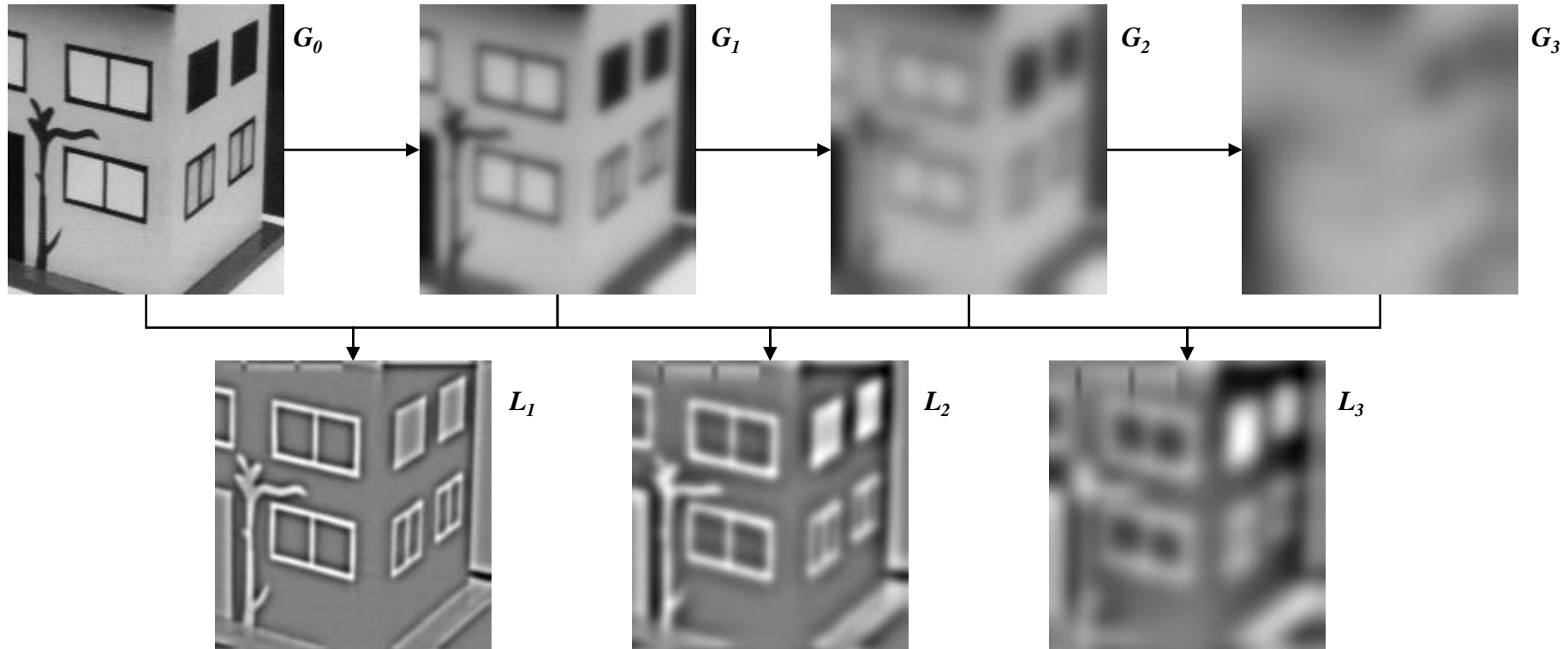


Détecteur de Harris multi-échelles



Points de Harris obtenus en calculant les dérivées premières par convolution avec une dérivée de gaussienne d'écart-type σ .

Détecteur SIFT : extrema dans l'espace d'échelle



La fonction $G_k(x,y) = G(x,y,k\sigma)$ est l'image convoluée par une gaussienne d'écart-type $k\sigma$. Les fonctions $L_k(x,y)$ correspondent à la différence (ici normalisée) entre 2 gaussiennes adjacentes.

La fonction $L_k(x,y)$ est une représentation laplacienne de l'image, qui correspond à une décomposition fréquentielle localisée : contribution des structures contrastées d'échelle (de « taille » $k\sigma$ au point (x,y)).

Les points sélectionnés par SIFT sont les maxima et les minima locaux de la fonction $L_k(x,y)$, à la fois dans l'échelle courante et dans les échelles adjacentes (voir ci-contre).

Points d'intérêt SIFT

Pour chaque extrema de l'espace d'échelle des différences de gaussiennes (point d'intérêt SIFT), on calcule la direction associée par :

$$\theta(x, y) = \arctan \left(\frac{G_y^\sigma(x, y)}{G_x^\sigma(x, y)} \right)$$

avec $G_x^\sigma(x, y) = \frac{\partial}{\partial x} G(x, y, \sigma) = I(x, y) * \frac{\partial}{\partial x} g_\sigma(x, y)$

(où σ est l'échelle sélectionnée)

Ci-contre, point d'intérêt SIFT : la direction de la flèche représente la direction θ et sa longueur l'échelle σ associée.

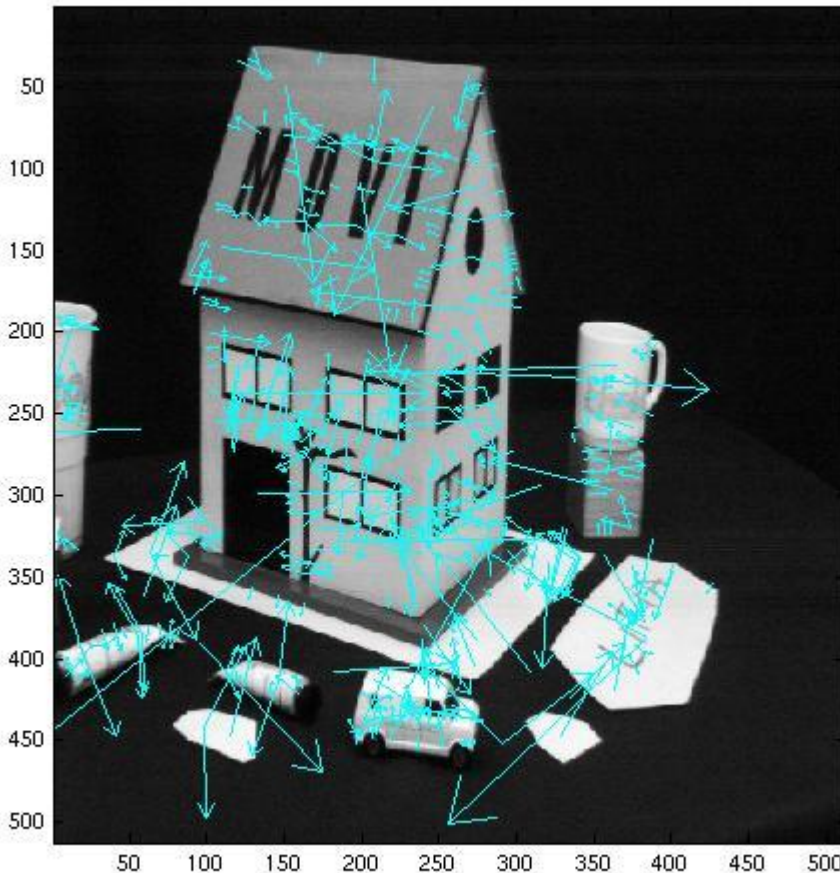
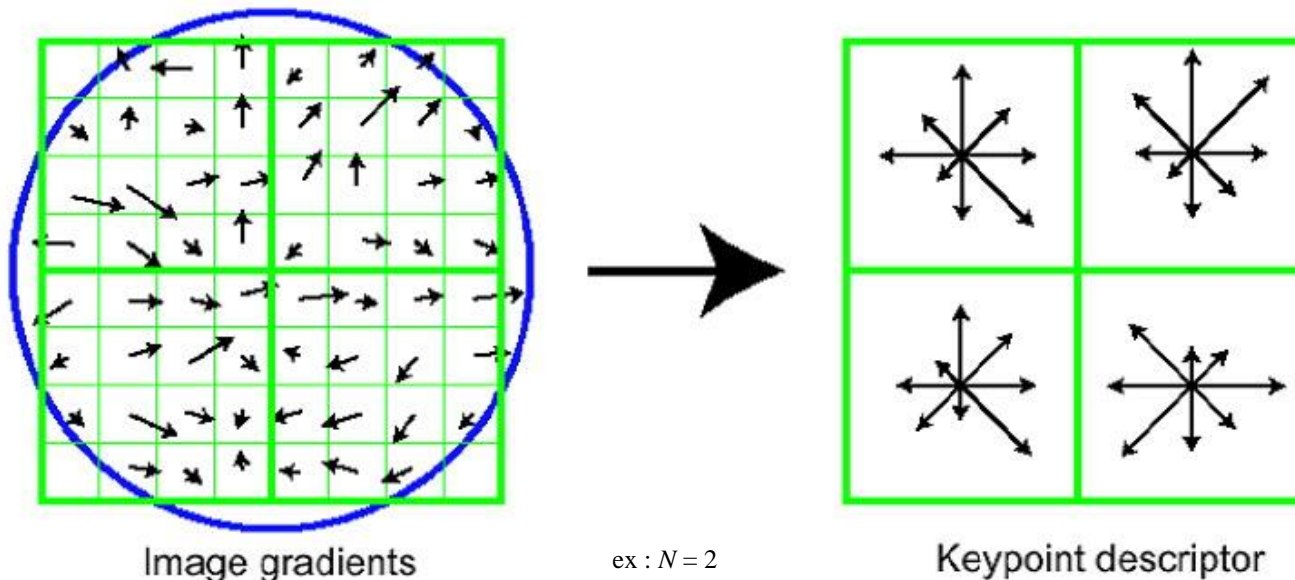


Image 1 : 589 points détectés.

Descripteur SIFT : histogramme d'orientation

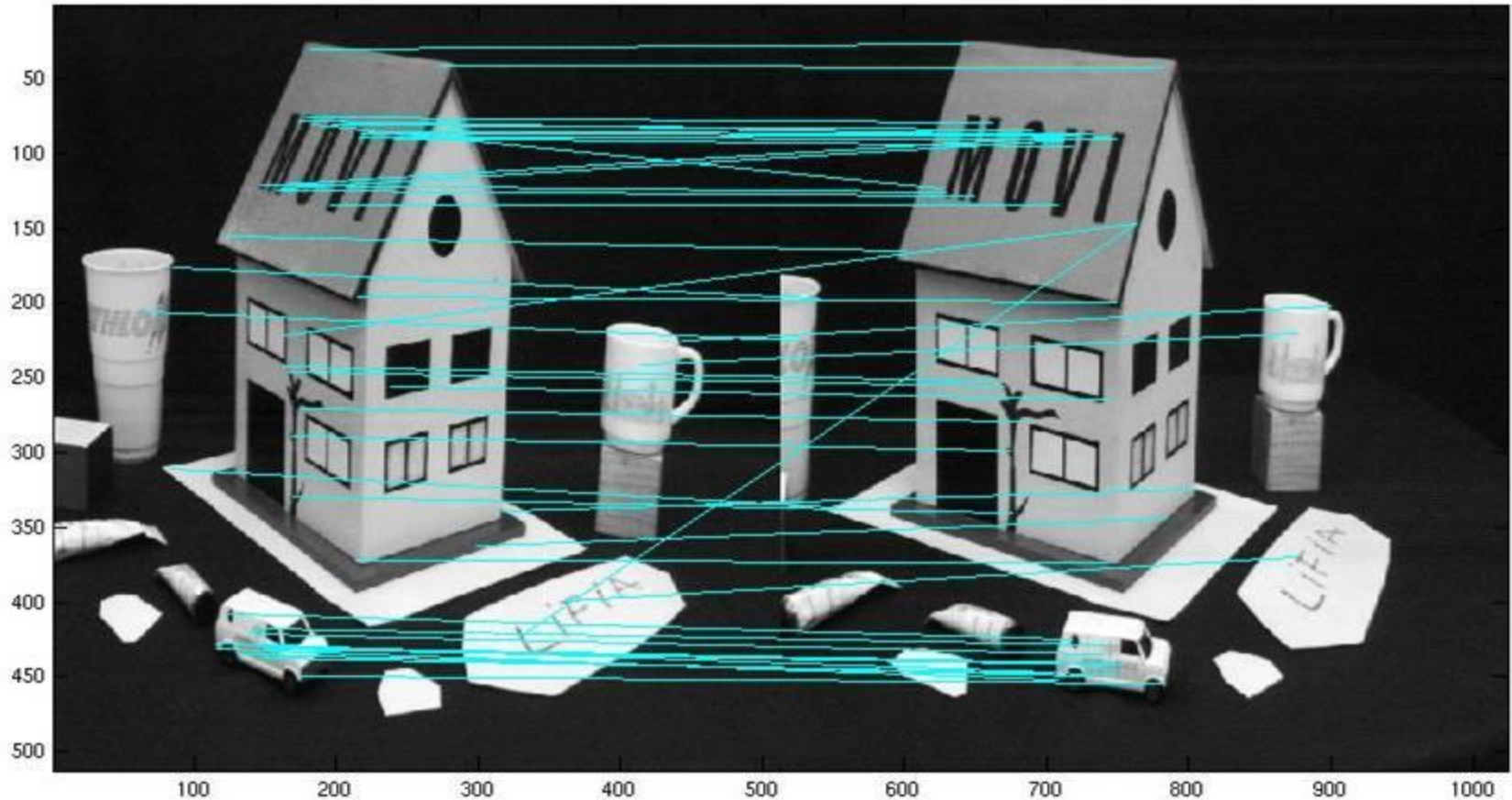
Les descripteurs associés aux points d'intérêt SIFT sont des histogrammes des orientations locales autour du point d'intérêt.

- On divise l'espace autour de chaque point d'intérêt (x,y) en N^2 carrés 4×4 .
- On calcule le gradient $(G_x(a,b,\sigma), G_y(a,b,\sigma))$ pour les $4 \times 4 \times N^2$ points (a,b) .
- Pour chaque carré 4×4 , on calcule un histogramme des orientations quantifiées en 8 directions, en pondérant par : (1) le module du gradient (2) l'inverse de la distance au point d'intérêt (x,y) .
- Pour être invariant en rotation : l'orientation locale du point d'intérêt $\theta(x,y)$ est utilisée comme *origine* (orientation nulle) des histogrammes.



Les descripteurs formés sont donc des vecteurs de taille $8 \times N^2$, qui seront appariés en utilisant une distance (e.g. distance euclidienne)

Appariement par SIFT



Résultat d'appariement par SIFT entre l'image (2) à gauche, 510 points détectés, et l'image (1) à droite, 589 points détectés. 51 points ont été appariés, ce qui correspond à une distance euclidienne entre les descripteurs en deçà d'un certain seuil.

Conclusion Chapitre 2

❑ RECALAGE : Mouvement global

- ❑ Corrélation de phase : *translation*

- ❑ Invariants de Fourier-Mellin : *rotation / homothétie*

❑ POURSUITE

❑ Points d'intérêt

- ❑ Harris : *points anguleux*

- ❑ SIFT : *extrema espace d'échelle*

❑ Descripteurs

- ❑ Invariants différentiels

- ❑ Histogrammes d'orientation

Bibliographie du Chapitre 2

- **H. Foroosh, J. Zerubia & M. Berthod 2002** « *Extension of phase correlation to subpixel registration* » *IEEE Transactions on Image Processing* 11(3) pp 188-200
- **Q. Chen, M. Defrise & F. Deconinck 1994** « *Symmetric Phase-Only Matched Filtering of Fourier-Mellin Transforms for Image Registration and Recognition* » *IEEE Transactions on Pattern Analysis and Machine Intelligence* 16(12) pp 1156-1168
- **C. Harris & M. Stephens 1988** « *A combined corner and edge detector* » *Alvey Vision Conference* pp 147-151
- **C. Schmid & R. Mohr 1997** « *Local grayvalue invariants for image retrieval* » *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19(5) pp 530-534
- **D.G. Lowe 2004** « *Distinctive Image Features from Scale-Invariant Keypoints* » *International Journal of Computer Vision* 60-2 pp 91-110