

TRAITEMENT D'IMAGES

Partie Introductive

Frédéric Cointault
Institut Agro Dijon
Responsable Equipe ATIP
UMR Agroécologie
26 Bd Dr Petitjean
21000 Dijon
+33 3 80 77 27 54

frederic.cointault@agrosupdijon.fr

L'INSTITUT NATIONAL D'ENSEIGNEMENT SUPÉRIEUR POUR L'AGRICULTURE, L'ALIMENTATION ET L'ENVIRONNEMENT



0 - Préambule

I - Introduction

II - Définitions

III - Pré-traitement des images

IV - Segmentation image et contours

V - Hough et morphologie mathématique

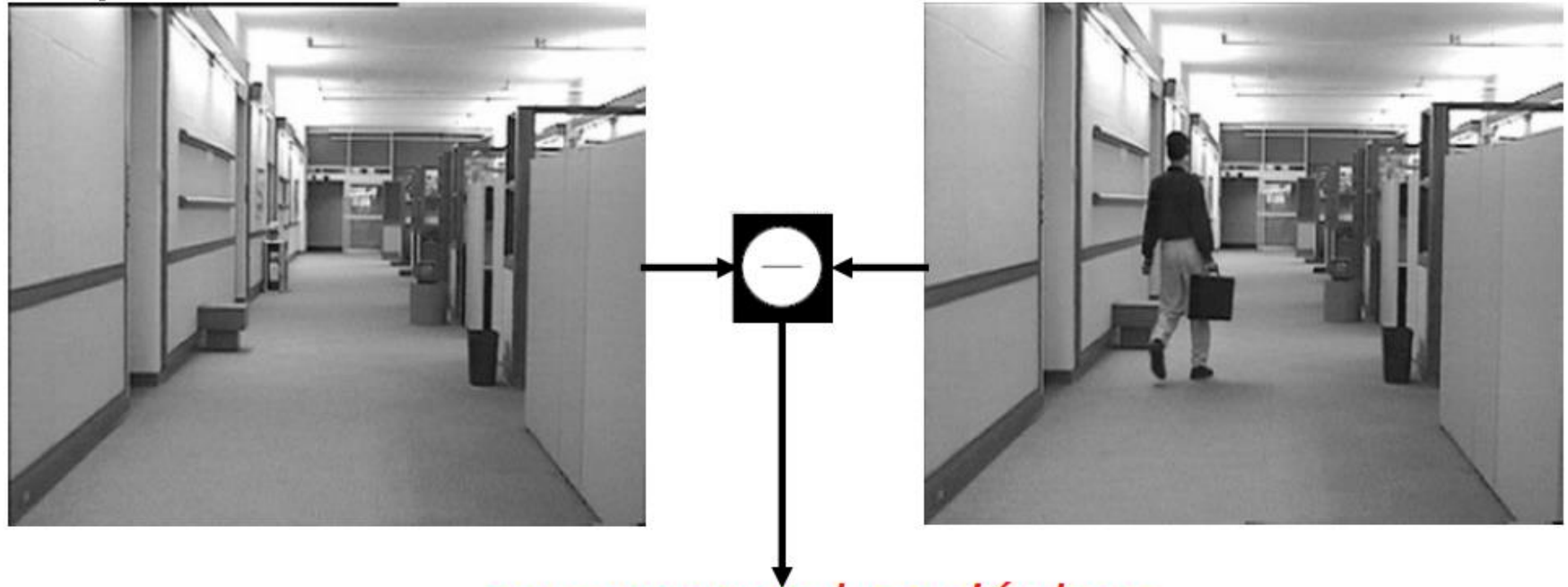
VI – Analyse et Reconnaissance de formes

VII – Détection de mouvement

VIII – Introduction au Deep Learning

VII – Détection de mouvement

Détection de mouvement :
différence
avec le fond



personne qui se déplace

Même si la différence d'images n'est pas la méthode la plus complète pour l'analyse du mouvement, elle est très rapide pour la détection simple du mouvement

VII – Détection de mouvement

Applications diverses

- Détection de personnes
 - Interaction homme-machine
 - Jeux vidéo - web cam
 - Détection de composantes sur un convoyeur
 - Compter les voitures sur une route
-
- Cas les plus simples
 - Utilisation d'un fond de couleur uniforme
 - Un fond quelconque mais statique

VII – Détection de mouvement

Application en réalité virtuelle



- Projection d'une personne dans un univers virtuel
 - Réalité virtuelle
 - Obtenu par soustraction d'arrière-plan

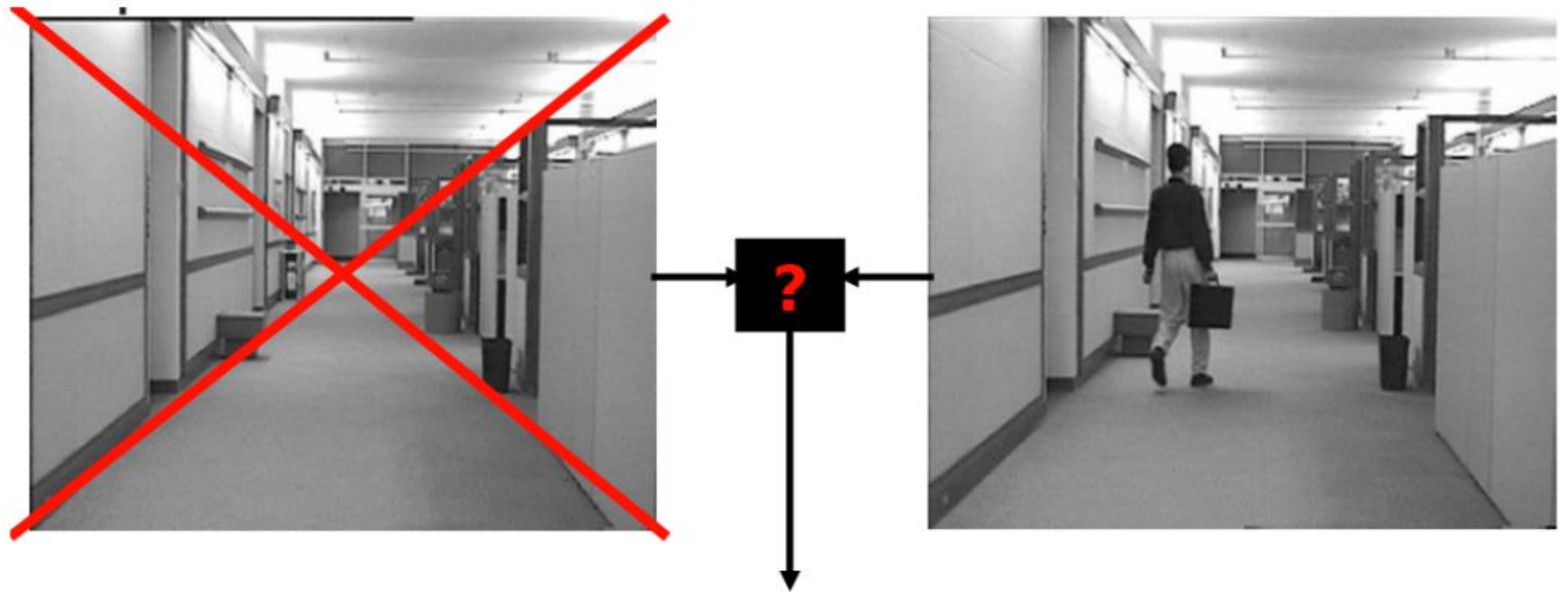
VII – Détection de mouvement

Détection de mouvement :
soustraction
d'arrière-plan

- Si on possède **une image de l'arrière-plan (AP)**, on soustrait l'image courante de cette image pour isoler l'objet en mouvement
 - L'image de l'arrière-plan ne varie pas en fonction du temps
- Algorithme de détection avec image d'arrière-plan :
 - **0** : $|I(x,y) - I_{AP}(x,y)| < \text{Seuil}$
 - **1** : $|I(x,y) - I_{AP}(x,y)| \geq \text{Seuil}$
- **On peut mettre à jour l'image d'arrière-plan**
 - Pour tenir compte des variations d'éclairage
 - Variation temporelle de l'image (objets déplacés, etc.)

VII – Détection de mouvement

Détection de mouvement :
construction
d'arrière-plan

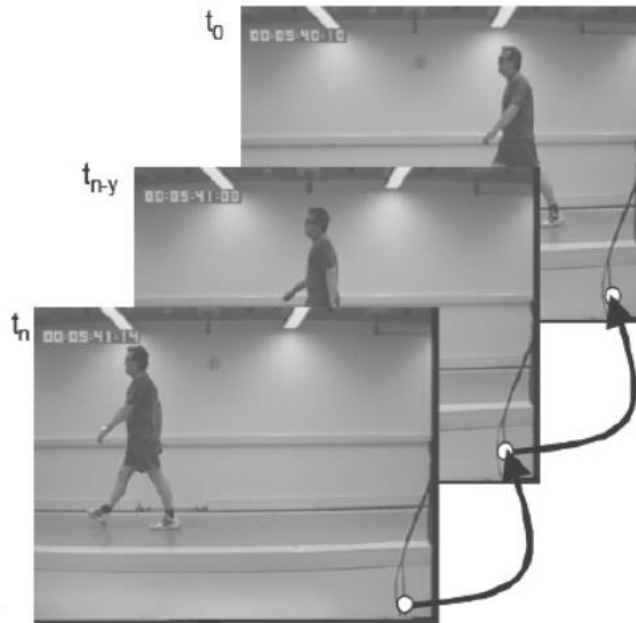


personne qui se déplace ?

Comment faire si on n'a pas l'image d'arrière-plan pour faire la détection ?

VII – Détection de mouvement

Détection de mouvement :
construction
d'arrière-plan



Séquence d'images

Tableau des valeurs
pour chaque pixel

$p(x1,y1) = [210, 118, \dots, 180]$

$p(x2,y2) = [87, 59, \dots, 86]$

...

$p(x_n,y_n) = [200, 200, \dots, 189]$

Arrière plan
Obtenu par:
Médiane,
Moyenne
récursive, ...



VII – Détection de mouvement

Détection de mouvement :
construction
d'arrière-plan

1- Moyenne temporelle récurrente:

$$\begin{aligned}M_t &= \alpha I_t + (1 - \alpha)M_{t-1} \\ &= \alpha(I_t - M_{t-1}) + M_{t-1}\end{aligned}$$

avec α compris entre 0 et 1

$$D_t = |\bar{M}_t - I_t|$$

VII – Détection de mouvement

Détection de mouvement :
construction
d'arrière-plan

2- Moyenne temporelle sigma-delta:

Pour chaque trame t
pour chaque pixel x :
$$\Delta_t(x) = M_t(x) - I_t(x)$$

(1)

Initialisation
pour chaque pixel x :
$$M_0(x) = I_0(x)$$

Pour chaque trame t
pour chaque pixel x :
if $\Delta_t(x) < 0$, $M_t(x) = M_{t-1}(x) + 1$
if $\Delta_t(x) > 0$, $M_t(x) = M_{t-1}(x) - 1$

(2)

VII – Détection de mouvement

Détection de mouvement :
post-
traitement



Image

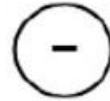


Image du fond

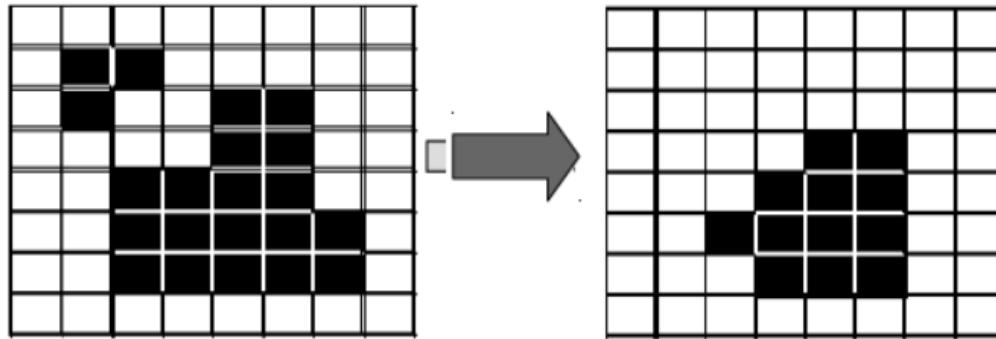


Détection

Après la détection, il reste encore du bruit, qu'il faut nettoyer pour obtenir l'objet qui nous intéresse

VII – Détection de mouvement

Détection de mouvement :
post-traitement



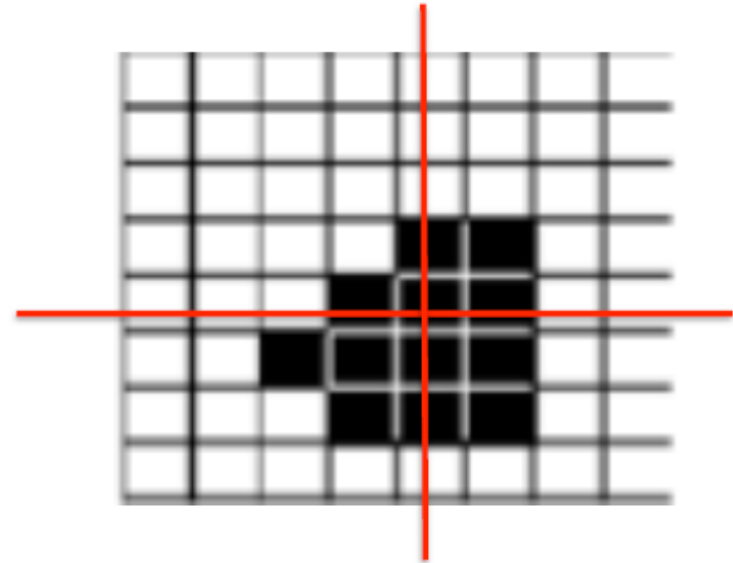
Erosion 3x3

Utilisation des opérateurs morphologiques comme l'érosion et la dilatation

VII – Détection de mouvement

Détection de mouvement :
post-traitement

Localisation et suivi
de la forme détectée

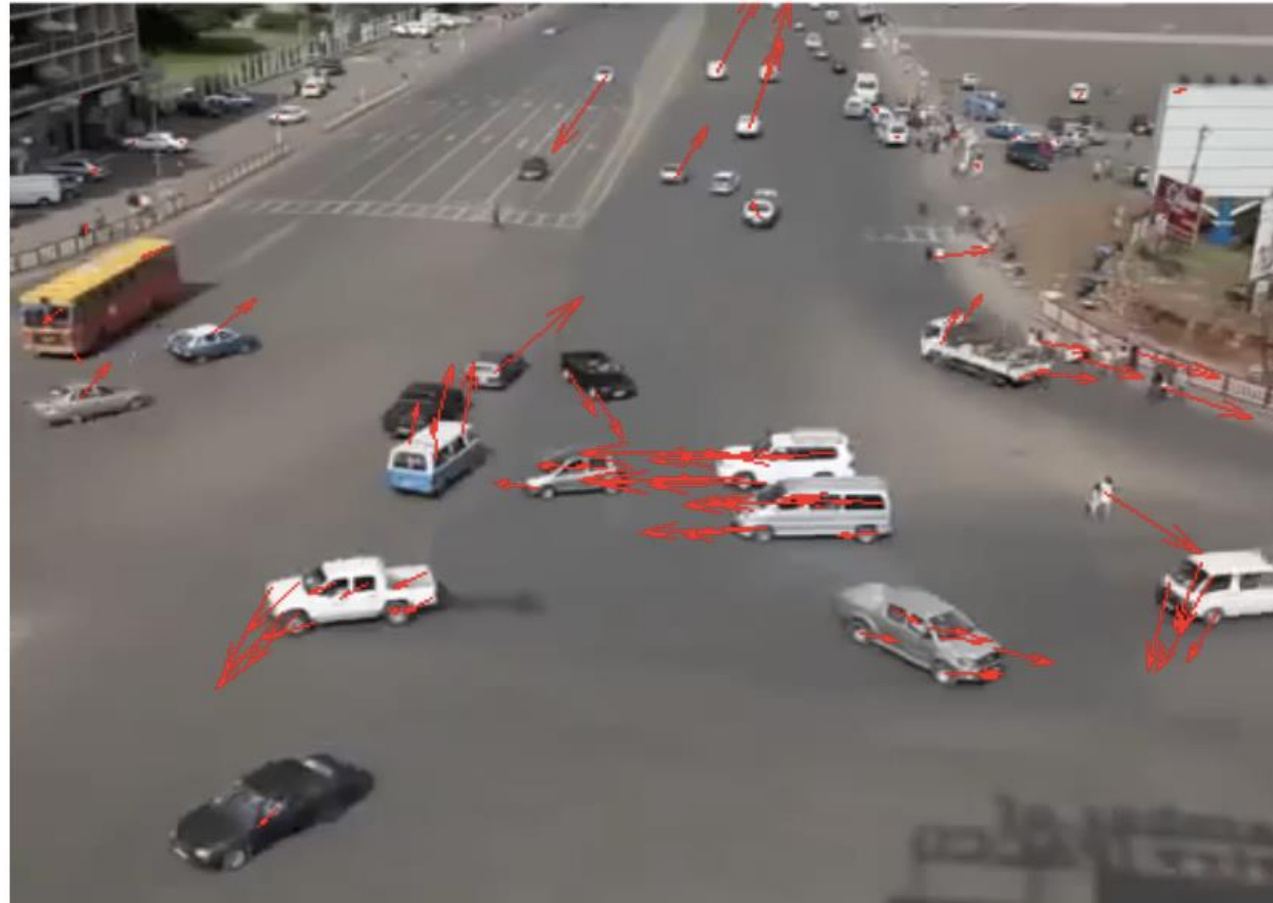


Calcul du centre de gravité

VII – Détection de mouvement

Flot optique: mesure de décalages faibles entre images

Estimation
du
mouvement

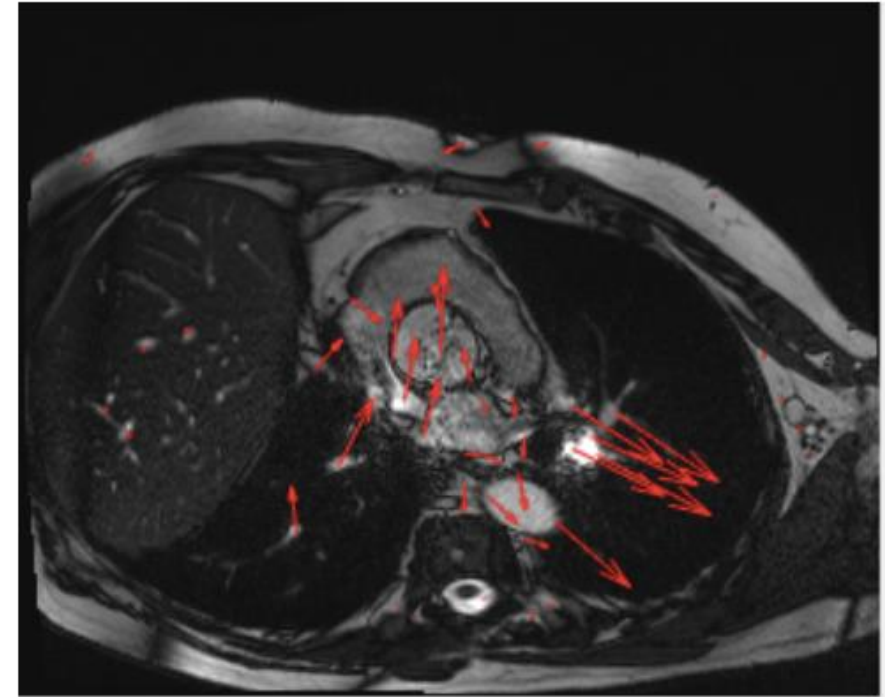


VII – Détection de mouvement

Flot optique : exemples



Marche



Mouvement Aorte

VII – Détection de mouvement

Considérons deux images $I_1(x,y)$ et $I_2(x,y)$ avec une translation de composantes v_x et v_y entre les deux images

$$I_2(x,y) = I_1(x - v_x, y - v_y)$$

Estimation du mouvement

En considérant une faible valeur de translation, on peut utiliser le développement de Taylor:

$$I_2(x,y) \cong I_1(x,y) - v_x \frac{\partial I_1(x,y)}{\partial x} - v_y \frac{\partial I_1(x,y)}{\partial y}$$

$$I_1(x,y) - I_2(x,y) \cong v_x \frac{\partial I_1(x,y)}{\partial x} + v_y \frac{\partial I_1(x,y)}{\partial y}$$

$$I_t(x,y) \cong \nabla I_1(x,y) \mathbf{v}$$

$$\text{Avec } \nabla I_1(x,y) = \left[\frac{\partial I_1(x,y)}{\partial x}, \frac{\partial I_1(x,y)}{\partial y} \right] \text{ et } \mathbf{v} = \begin{bmatrix} v_x \\ v_y \end{bmatrix}$$

VII – Détection de mouvement

UNE SOLUTION : LUKAS-KANADE

Hypothèse : même translation pour tous les pixels

⇒ unique vecteur \mathbf{v} pour tous les pixels.

⇒ construction d'un système surdéterminé de type $A\mathbf{v}=\mathbf{b}$

Estimation
du
mouvement

$$A = \begin{pmatrix} \frac{\partial I_1}{\partial x}(p_1) & \frac{\partial I_1}{\partial y}(p_1) \\ \vdots & \vdots \\ \frac{\partial I_1}{\partial x}(p_n) & \frac{\partial I_1}{\partial y}(p_n) \end{pmatrix}, \mathbf{v} = \begin{bmatrix} v_x \\ v_y \end{bmatrix}, \mathbf{b} = - \begin{pmatrix} \frac{\partial I_1}{\partial t}(p_1) \\ \vdots \\ \frac{\partial I_1}{\partial t}(p_n) \end{pmatrix}$$

Avec p_i le $i^{\text{ème}}$ pixel de l'image et n le nombre de pixels de l'image.

Résolution du problème par moindres-carrés : $A^T A \mathbf{v} = A^T \mathbf{b}$

VII – Détection de mouvement

UNE SOLUTION : LUKAS-KANADE

$$A^T A = \begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{bmatrix} \text{ et } A^T b = \begin{bmatrix} \sum I_t I_x \\ \sum I_t I_y \end{bmatrix}$$

$$\text{Avec } I_x = \frac{\partial I_1}{\partial x}, I_y = \frac{\partial I_1}{\partial y}, I_t = \frac{\partial I_1}{\partial t} = I_2 - I_1$$

Solution : la matrice $A^T A$ doit être inversible. La solution est alors : $(A^T A)^{-1} A^T b$.

$$\text{Soit } \widehat{v}_x = \frac{1}{\text{Det}} \left(\sum I_t I_x \sum I_y^2 - \sum I_t I_y \sum I_y I_x \right)$$

$$\widehat{v}_y = \frac{1}{\text{Det}} \left(\sum I_t I_y \sum I_x^2 - \sum I_t I_x \sum I_y I_x \right)$$

$$\text{Avec } \text{Det} = \sum I_y^2 \sum I_x^2 - \left(\sum I_x I_y \right)^2$$

Estimation
du
mouvement

VII – Détection de mouvement

CALCUL DES GRADIENTS SPATIAUX

Différence finie à droite

$$I_x(x, y) = I(x + 1, y) - I(x, y)$$

$$I_y(x, y) = I(x, y + 1) - I(x, y)$$

$I(x, y)$	$I(x+1, y)$
$I(x, y+1)$	$I(x+1, y+1)$

On peut également utiliser :

$$I_x(x, y) = \frac{(I(x + 1, y) - I(x, y)) + (I(x + 1, y + 1) - I(x, y + 1))}{2}$$

$$I_y(x, y) = \frac{(I(x, y + 1) - I(x, y)) + (I(x + 1, y + 1) - I(x + 1, y))}{2}$$

Besoin de recalage d'un demi-pixel :

$I_1(x, y) \leftarrow I_1(x + 0.5, y + 0.5)$ avec une interpolation bilinéaire

Soit $I_1(x, y) \leftarrow \frac{I_1(x, y) + I_1(x+1, y) + I_1(x, y+1) + I_1(x+1, y+1)}{4}$

On réalise ceci sur I1 et I2.

Estimation
du
mouvement

VII – Détection de mouvement

Estimation du mouvement

Implementing Lucas Kanade Method

for each point, calculate I_x , I_y , I_t

```
Ix_m = conv2(im1, [-1 1; -1 1], 'valid'); % partial on x
Iy_m = conv2(im1, [-1 -1; 1 1], 'valid'); % partial on y
It_m = conv2(im1, ones(2), 'valid') + conv2(im2, -ones(2), 'valid');
u = zeros(length(C),1);
v = zeros(length(C),1);

% within window ww * ww
for k = 1:length(C(:,2))
    i = C(k,2);
    j = C(k,1);
    Ix = Ix_m(i-w:i+w, j-w:j+w);
    Iy = Iy_m(i-w:i+w, j-w:j+w);
    It = It_m(i-w:i+w, j-w:j+w);

    Ix = Ix(:);
    Iy = Iy(:);
    b = -It(:); % get b here

    A = [Ix Iy]; % get A here
    nu = pinv(A)*b;

    u(k)=nu(1);
    v(k)=nu(2);
end;
```