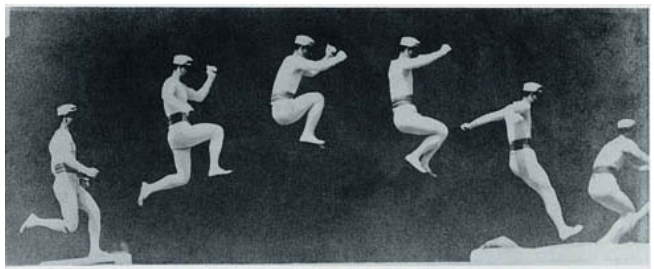
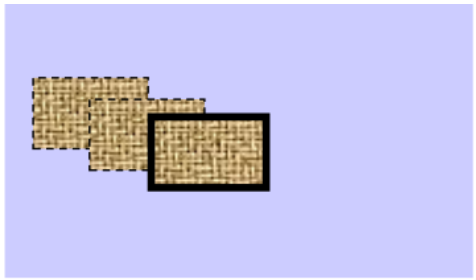
UPMC - Sorbonne Université
Master Imagerie - UE VISION
Suivi d'objets dans les vidéos

Antoine Manzanera
ENSTA-Paris



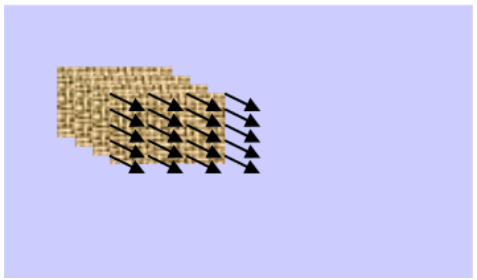
Positionnement dans l'UE VISION

Trois catégories de traitement d'images liés à l'analyse du mouvement :



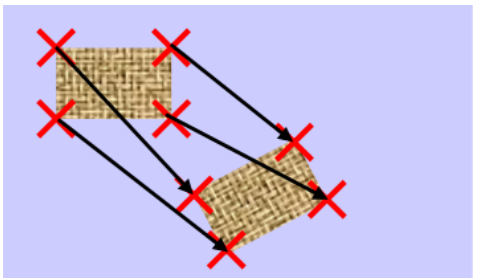
Détection

Séparer les pixels des objets mobiles et du fond statique



Estimation

Calculer la vitesse apparente de chaque pixel



Poursuite

Apparier certaines structures d'une image à l'autre

Contenu et Objectifs du Cours

- Vocabulaire : *suivi* = *poursuite* = *tracking*
- Présenter les caractéristiques, enjeux et difficultés du suivi d'objets dans les séquences d'images.
- Présenter les composantes d'un algorithme de suivi, ainsi que les principales familles de méthodes utilisées pour le suivi d'objets.
- Expliquer les principes d'*observation* et de *prédiction* fondant les différentes approches de suivi d'objets.

Plan du Cours

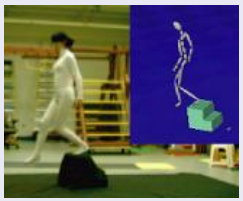
Domaines d'application

Vidéosurveillance

- Accès
- Comportement
- Reconnaissance

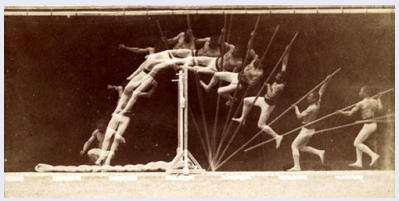
Interaction Homme-Machine

- Commande visuelle
- Contrôle avatar
- Langue des signes



Assistance

- Analyse de la marche
- Aide à la personne
- Analyse sportive



Suivi d'objets

Contexte

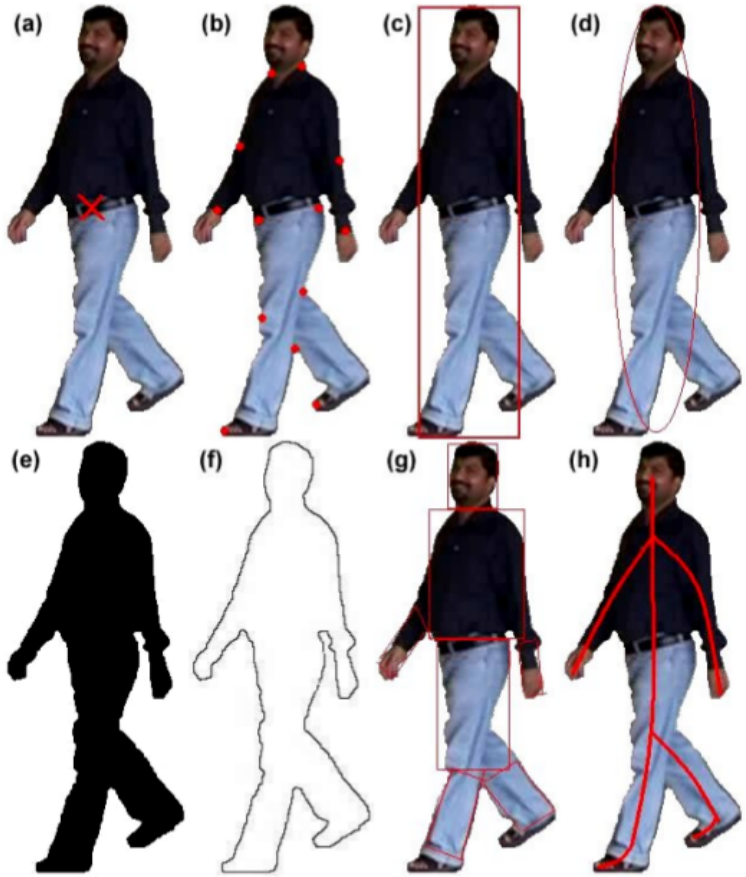
- Caméra mobile
- Objets mobiles et déformables

Localisation ou Segmentation ?

L'objectif du suivi est de localiser un objet défini initialement (comment ?) en calculant dans chaque image de la vidéo le support spatial (lequel ?) englobant l'objet.

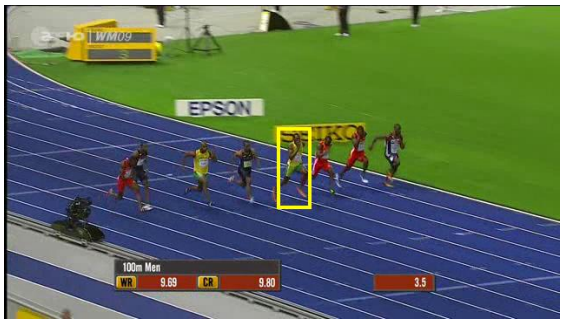
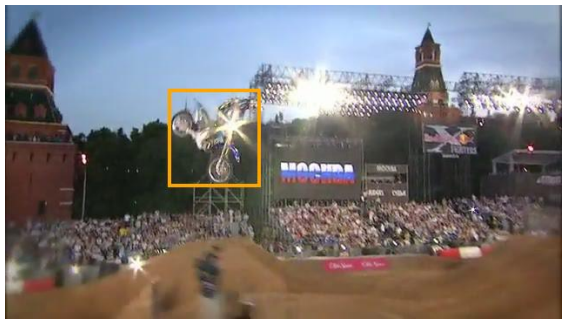
Support spatial de l'objet

Le support de représentation de l'objet est déterminant en termes de : Flexibilité / Fidélité / Complexité / Invariance...
Ci-contre quelques exemples typiques : (a) Un point, (b) Plusieurs points, (c) Rectangle englobant, (d) Ellipse, (e) Silhouette, (f) Contour, (g) Parties, (h) Squelette.



Tiré de **[Jalal12]**

Suivi : difficultés et challenges

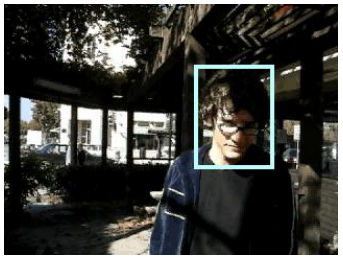




Exemples issus de la base de données de la compétition **[VOT14]**.

- Rotations, changements d'échelle
- Objets 3d, déformables
- Objets similaires
- Fond complexe et variable
- Occultations
- Changement d'illumination
- Faible contraste
- Flous de bougé
- Mouvements brusques

Suivi : difficultés et challenges



Exemples issus de la base de données de la compétition **[VOT14]**.

- Rotations, changements d'échelle
- Objets 3d, déformables
- Objets similaires
- Fond complexe et variable
- Occultations
- Changement d'illumination
- Faible contraste
- Flous de bougé
- Mouvements brusques

Composants fondamentaux du suivi

Un algorithme de suivi est composés de deux éléments fondamentaux :

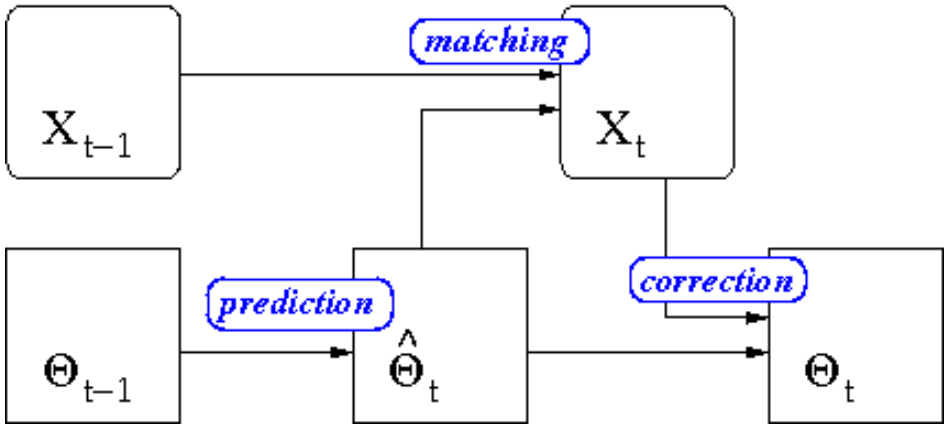
- **Prédiction** : Il s'agit, pour chaque image de la séquence, de fournir une *hypothèse* initiale de localisation de l'objet suivi. Cette hypothèse est en général fondée sur un *modèle dynamique* (ex : cinématique) de l'objet et de la scène (caméra).
- **Détection** : Il s'agit de localiser l'objet à partir d'*observations* extraites de l'image. Cette recherche est en général fondée sur un *modèle d'apparence*.

Formalisme général

Le suivi est souvent formalisé sous la forme générale suivante :

$$\arg\max_{\theta \in S} P(\Theta_t = \theta / \Theta_{t-1} = \theta', X_t = x)$$

où S représente un espace d'*états* ; θ est un paramètre d'état multidimensionnel, qui peut inclure la position, la vitesse, l'orientation, l'échelle, etc, de l'objet suivi ; X_t représente un vecteur d'observation extrait de l'espace image.



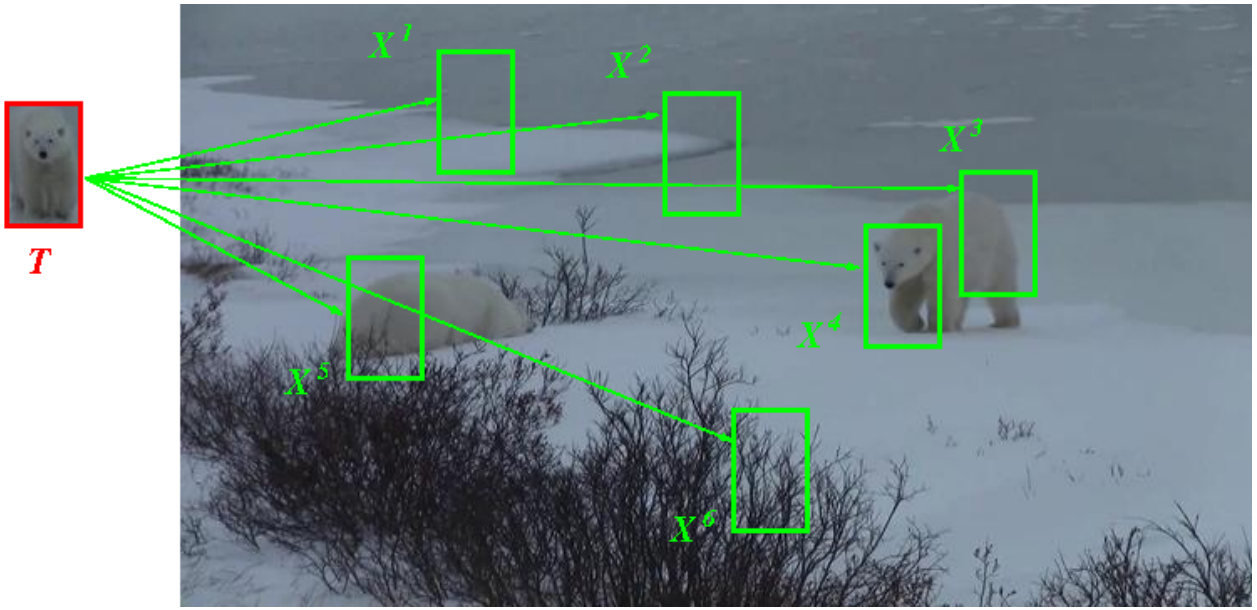
Prédiction *vs* Détection

L'équilibre entre prédiction et détection est très variable selon le type d'algorithmes, et peut même être radicalement défait en faveur de l'une ou l'autre :

- **Tracking by detection:** Le suivi est entièrement à la charge du détecteur : pas d'hypothèse de localisation requise.
- **Track before detect:** Le suivi est essentiellement assuré par la prédiction et peut donc fonctionner en cas de dégradation extrême de la visibilité (occultation, bruit, petite taille...).

Mesures de (di)similarité globale

Les mesures de (di)similarité globale s'applique entre deux vecteurs T et X de même dimension n, dont l'un représente le modèle (template) de l'objet sous la forme (en général) d'une imagette rectangulaire, et l'autre une imagette extraite de l'image courante correspondant à une hypothèse de localisation.



Mesures de disimilarité globale

Distances de Minkowski

$$\mathcal{D}_k(T,X) = \sum_{i<n} |T_i - X_i|^k$$

Les distances SAD $\mathcal{D}_1$ ou SSD $\mathcal{D}_2$ sont très courantes pour apparier des imagettes.

Ce sont des mesures de dissimilarité dont la valeur dépend fortement de la taille n et du contenu des images.

L'algorithme de détection consiste donc à parcourir un espace d'hypothèses de localisation $\{X^h\}_{h\in\mathcal{H}}$ pour trouver celle qui minimise la disimilarité :

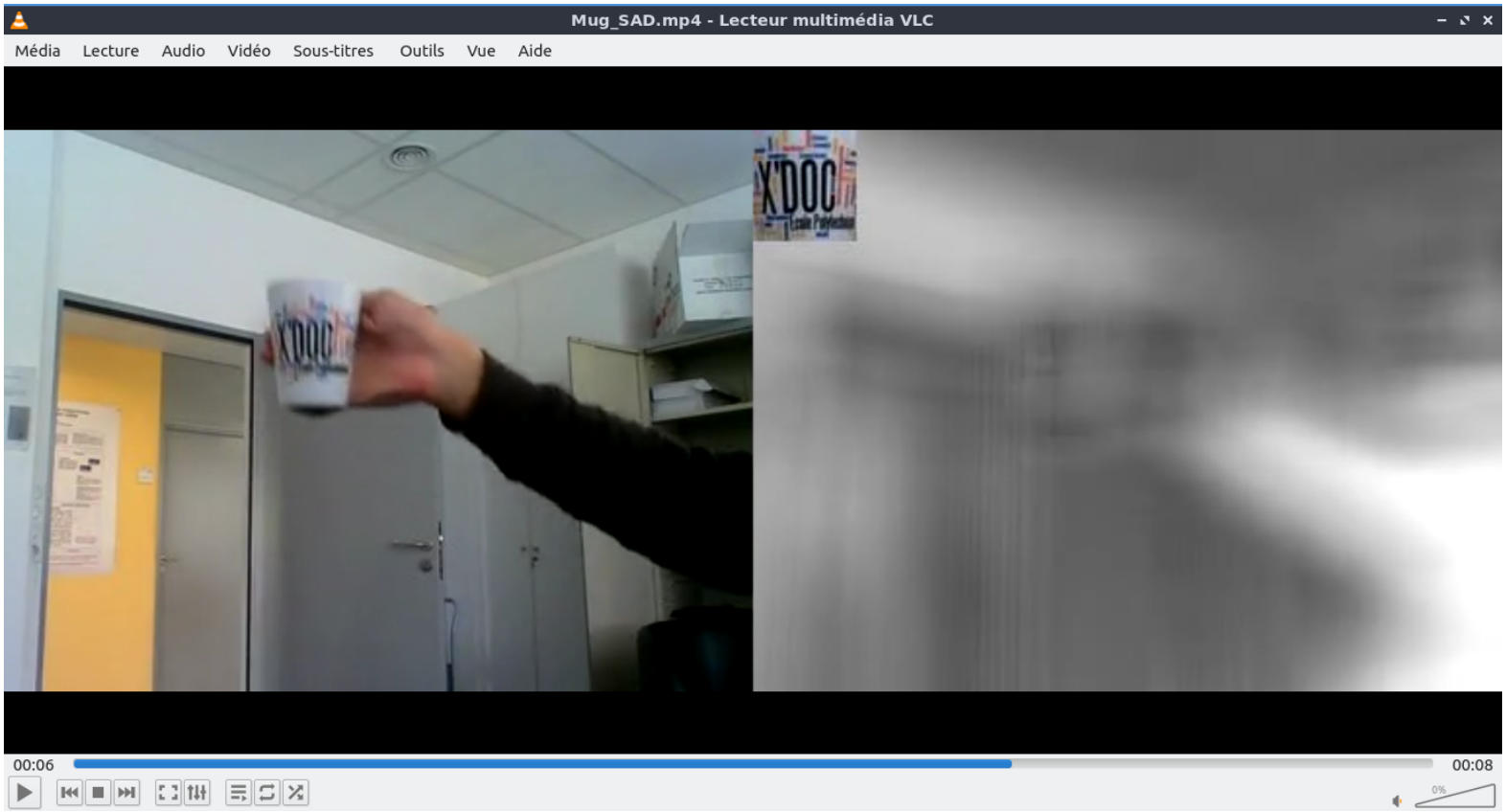
$$l = \arg\min_{h\in\mathcal{H}} \mathcal{D}_k(T, X^h)$$

\+ Modèle adaptable temporellement

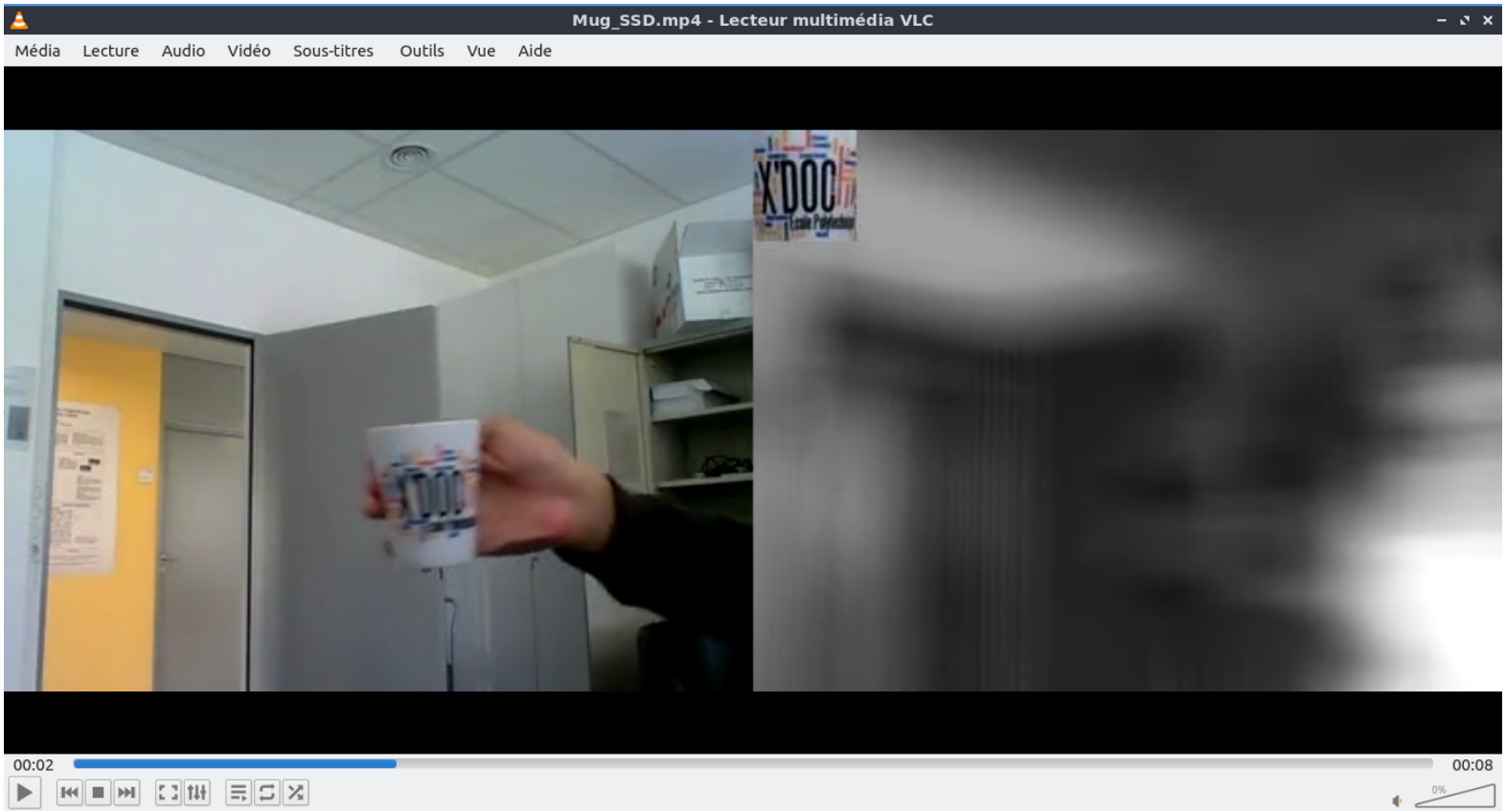
\- Sensible aux déformations

\- Sensible aux variations d'illumination

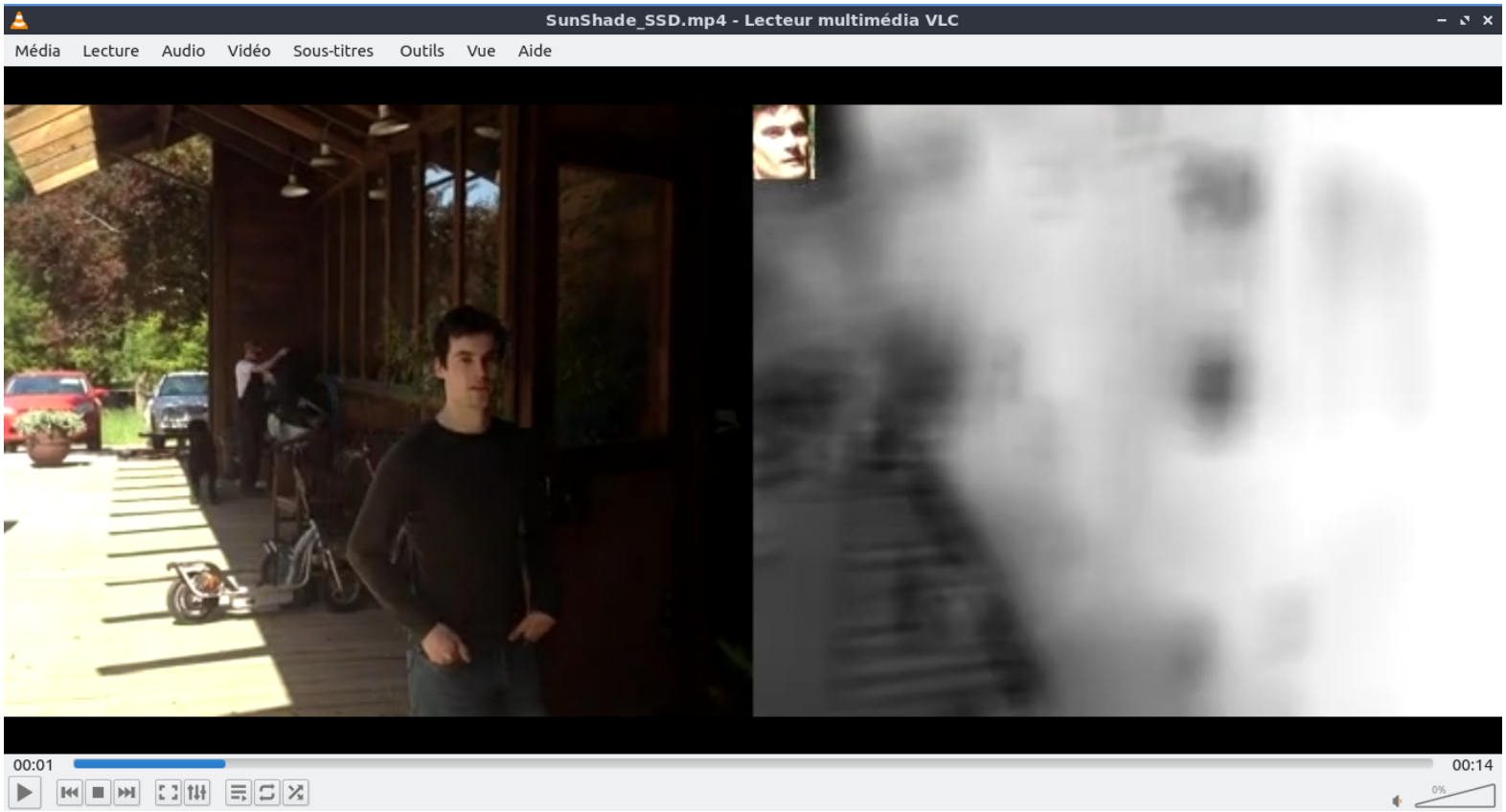
Suivi par Minimisation de SAD



Suivi par Minimisation de SSD



Suivi par Minimisation de SSD



Mesures de similarité globale

Coefficient de corrélation

Le produit scalaire normalisé des deux imagettes centrées varie entre -1 (dissimilarité totale) et +1 (identité). Soit $\tilde{X}$ la valeur moyenne de X sur son support.

$$\chi(T,X) = \frac{\sum_{i<n}(T_i - \tilde{T})(X_i - \tilde{X})}{\sqrt{\sum_{i<n}(T_i - \tilde{T})^2}\sqrt{\sum_{i<n}(X_i - \tilde{X})^2}}$$

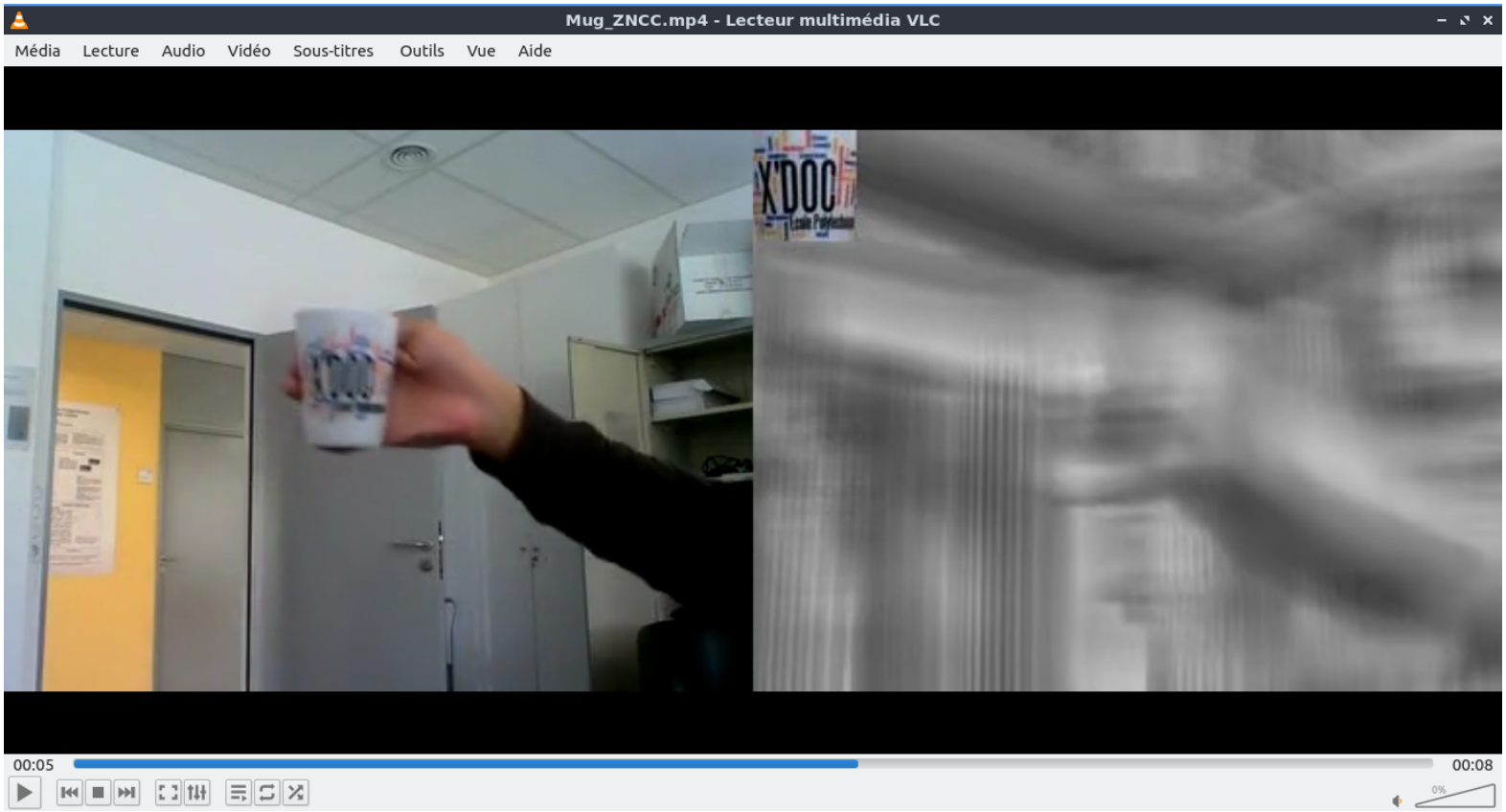
L'algorithme de détection consiste donc à parcourir un espace d'hypothèse de localisation $\{X^h\}_{h\in\mathcal{H}}$ pour trouver celle qui maximise la corrélation :

$$I = \arg\max_{h\in\mathcal{H}} \chi(T, X^h)$$

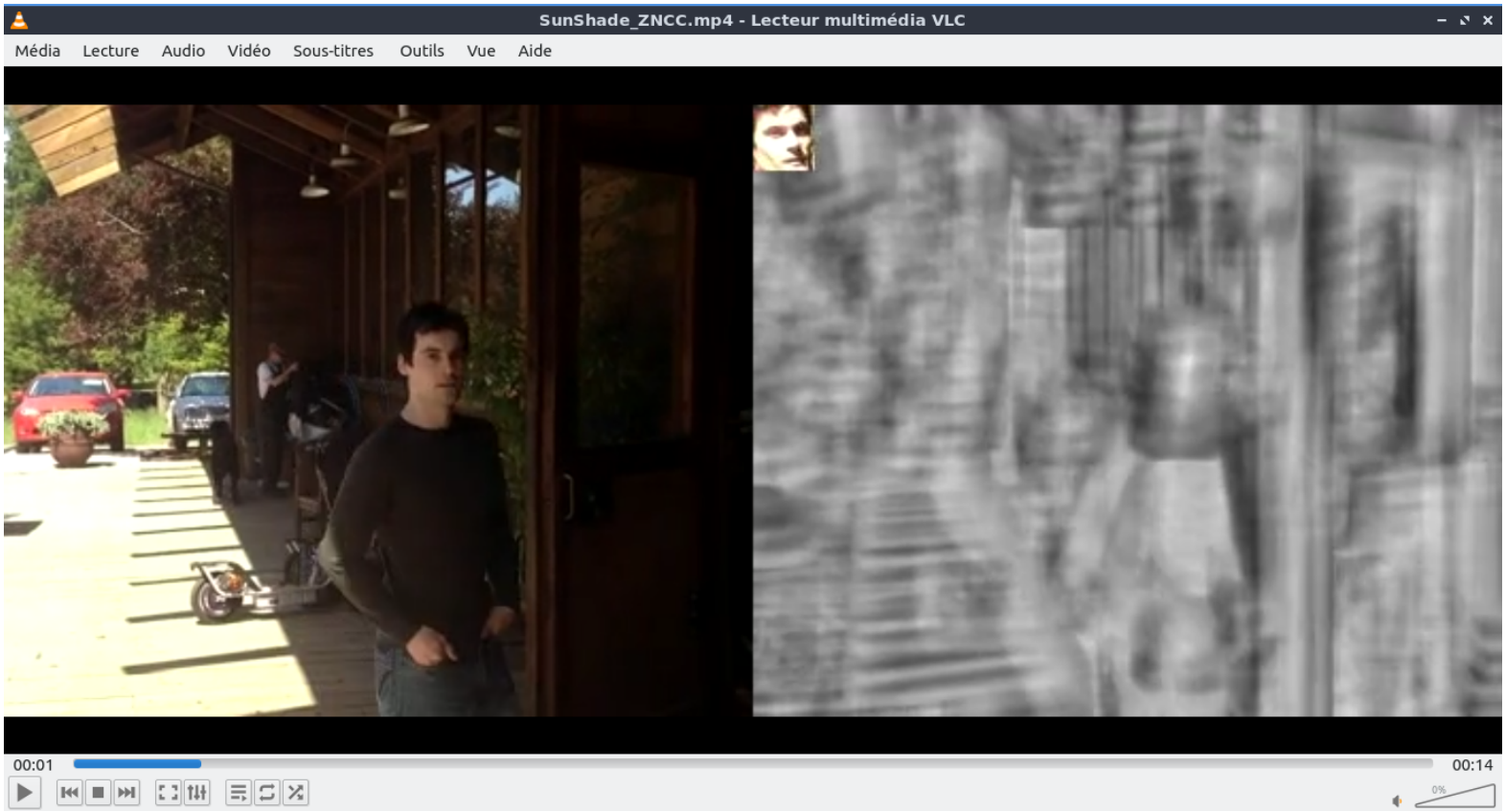
\+ Robuste aux variations d'illumination

\- Sensible aux déformations

Suivi par Maximisation de ZNCC



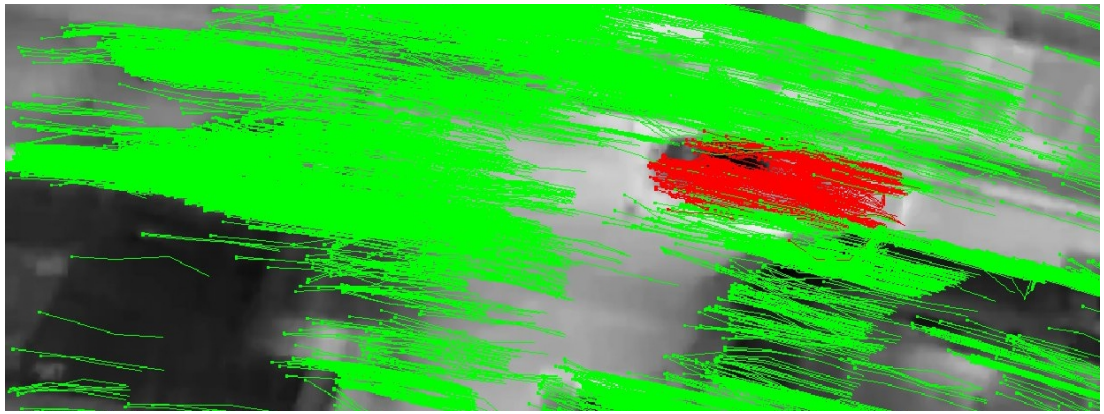
Suivi par Maximisation de ZNCC



Champ de déplacements apparents

Un algorithme de flux optique ou de suivi de points (voir cours "Estimation") peut estimer le déplacement apparent de points $\{\mathbf{t}_j\}_j \in T$ du modèle dans l'image courante, en appariant localement chaque point $\mathbf{t}_j$ à son homologue $\mathbf{y}_j$ dans l'image. La détection peut alors être fournie par une statistique sur les points homologues $Y = \{\mathbf{y}_j\}$, par exemple le médian :

$$I = \arg \min_{\mathbf{m} \in Y} \sum_{\mathbf{y} \in Y} ||\mathbf{m} - \mathbf{y}||$$



Similarité entre distributions

Index de *Bhattacharyya*

Cet index, compris entre 0 and 1 (égalité parfaite) mesure la similarité entre deux distributions. Soit H_T et H_X les histogrammes normalisés associés respectivement au modèle et à une hypothèse de localisation :

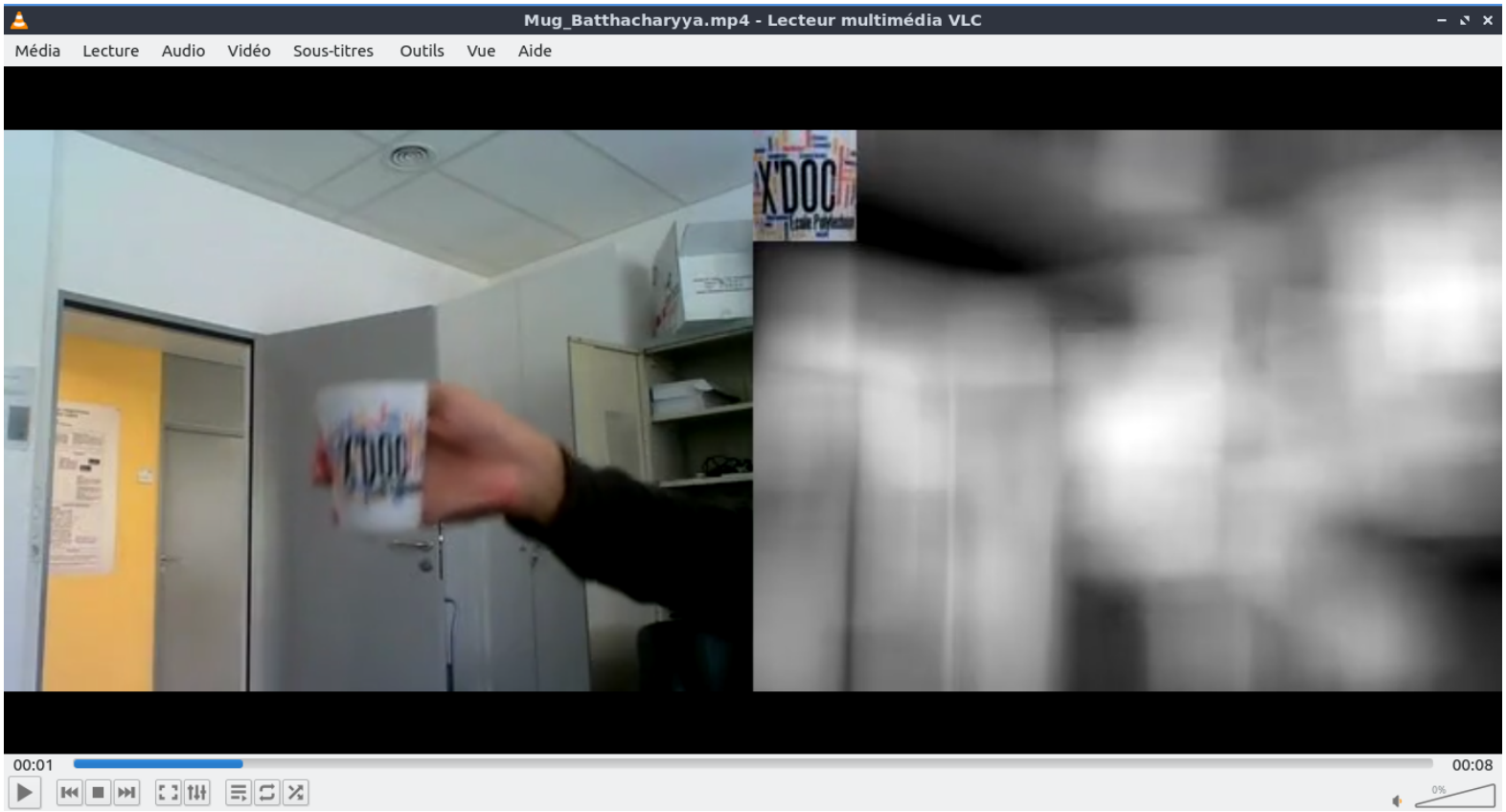
$$\mathcal{B}(H_T, H_X) = \sum_v \sqrt{H_T(v) H_X(v)}$$

\+ Robuste à tout type de déformation

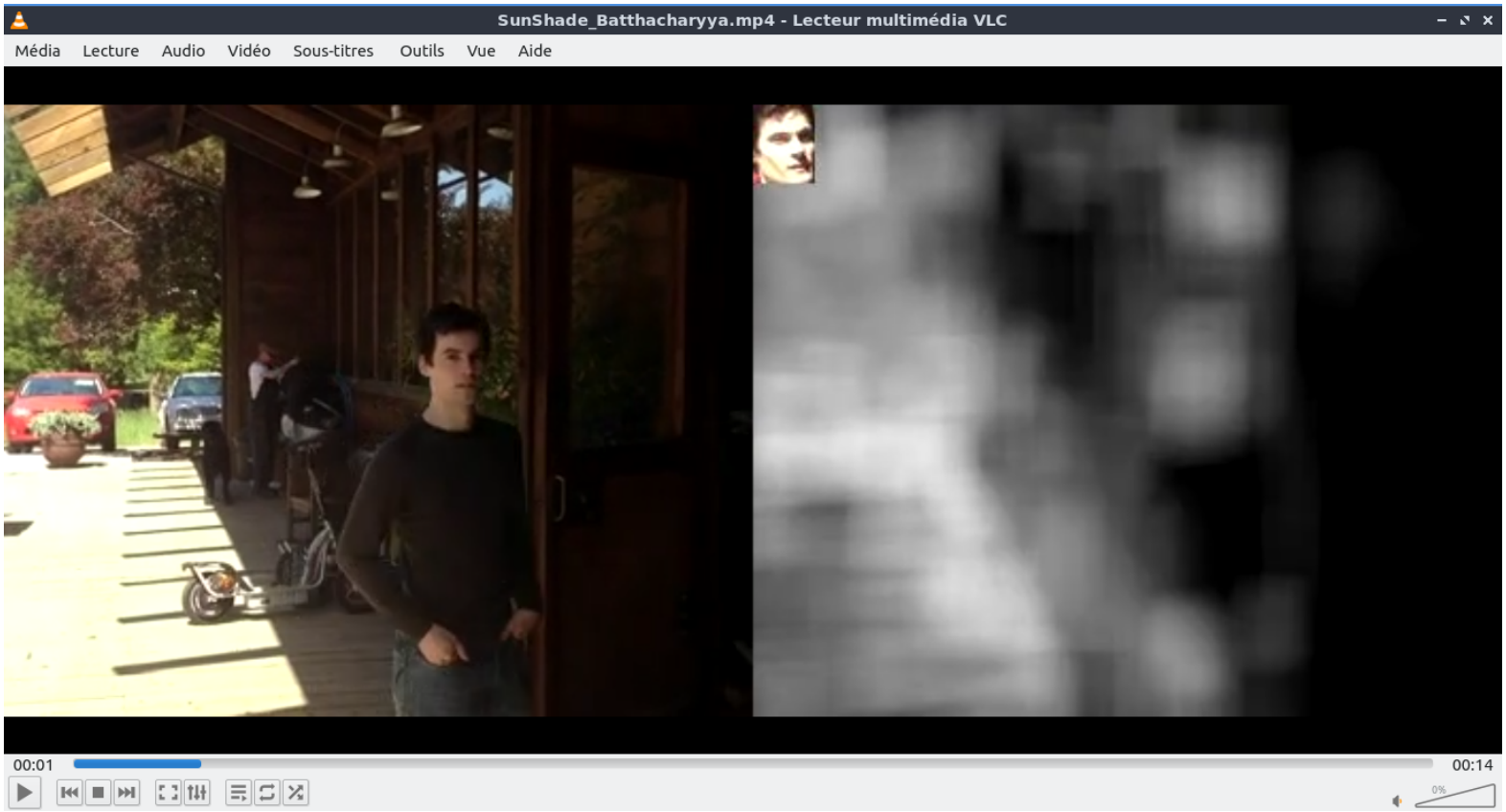
\- Sensible aux variations d'illumination

\- Peu discriminant sur le plan géométrique

Suivi de distribution



Suivi de distribution



Transformées de Hough Généralisées

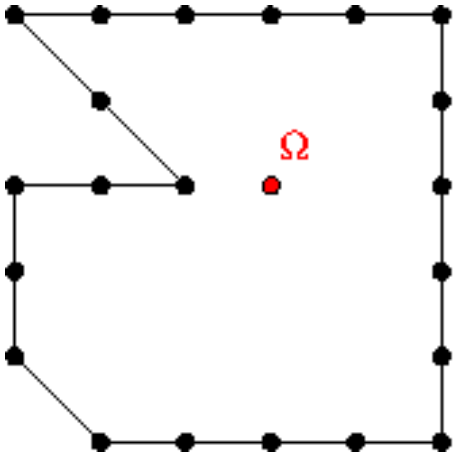
Les transformées de Hough généralisées (dites aussi "Modèles implicites de forme") sont une représentation d'objet fondée sur la *coocurrence d'éléments locaux*.

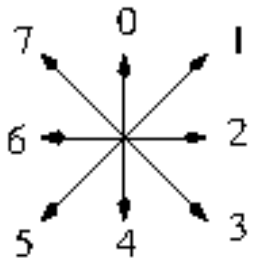
Modélisation

- Le modèle (template) T est échantillonné par un ensemble de points $\{\mathbf{t}_i\}_i$.
- Pour chaque point $\mathbf{t}_i$, on calcule :
  - un index d'apparence $\lambda(\mathbf{t}_i)$ (ex : couleur, orientation, courbure...).
  - un vecteur $\mathbf{v}(\mathbf{t}_i)$ correspondant à la position relative de $\mathbf{t}_i$ par rapport au centre du template T.
  - (éventuellement) un indice de confiance $\omega(\mathbf{t}_i)$.
- On indexe l'ensemble des triplets précédents (tableau, arbres de recherche, arbres aléatoires...) par l'index d'apparence λ.

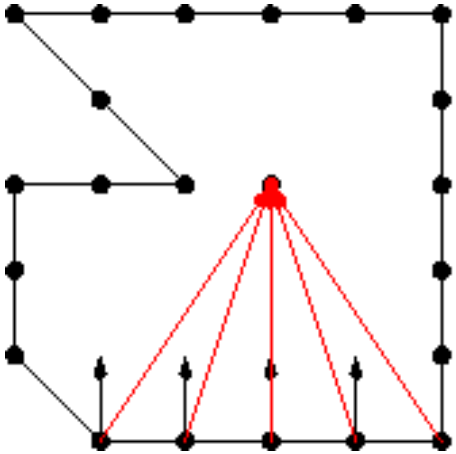
Construction de la R-Table

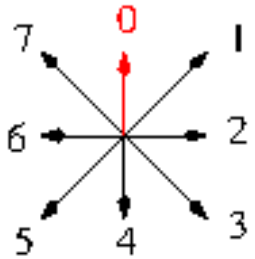
La R-Table est un modèle de forme, construit à partir d'un prototype. Soit Ω le centre arbitraire du prototype. Chaque point M du prototype est indexé par une caractéristique géométrique i, correspondant à l'indice de lignes de la R-table. Celle-ci est construite en ajoutant à la ligne d'indice i le vecteur $\overrightarrow{M\Omega}$, correspondant à la position du point par rapport au centre.
Par exemple soit le contour suivant vu comme un prototype, indexé par la direction normal au contour, quantifiée à 8 valeurs :





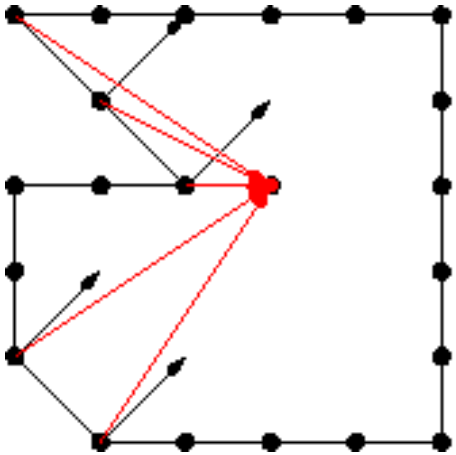
Construction de la R-Table

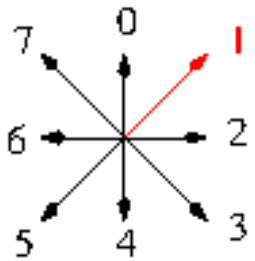




Indice	Liste des vecteurs					
0	$\begin{pmatrix}-2\\-3\end{pmatrix}$	$\begin{pmatrix}-1\\-3\end{pmatrix}$	$\begin{pmatrix}0\\-3\end{pmatrix}$	$\begin{pmatrix}1\\-3\end{pmatrix}$	$\begin{pmatrix}2\\-3\end{pmatrix}$	end

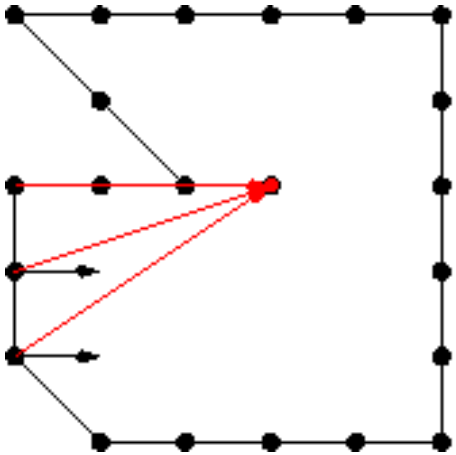
Construction de la R-Table

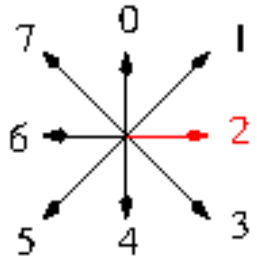




Indice	Liste des vecteurs					
0	$\begin{pmatrix}-2\\-3\end{pmatrix}$	$\begin{pmatrix}-1\\-3\end{pmatrix}$	$\begin{pmatrix}0\\-3\end{pmatrix}$	$\begin{pmatrix}1\\-3\end{pmatrix}$	$\begin{pmatrix}2\\-3\end{pmatrix}$	end
1	$\begin{pmatrix}2\\-3\end{pmatrix}$	$\begin{pmatrix}3\\-2\end{pmatrix}$	$\begin{pmatrix}1\\0\end{pmatrix}$	$\begin{pmatrix}2\\1\end{pmatrix}$	$\begin{pmatrix}3\\2\end{pmatrix}$	end

Construction de la R-Table



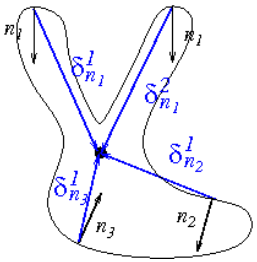


Indice	Liste des vecteurs					
0	$\begin{pmatrix}-2\\-3\end{pmatrix}$	$\begin{pmatrix}-1\\-3\end{pmatrix}$	$\begin{pmatrix}0\\-3\end{pmatrix}$	$\begin{pmatrix}1\\-3\end{pmatrix}$	$\begin{pmatrix}2\\-3\end{pmatrix}$	end
1	$\begin{pmatrix}2\\-3\end{pmatrix}$	$\begin{pmatrix}3\\-2\end{pmatrix}$	$\begin{pmatrix}1\\0\end{pmatrix}$	$\begin{pmatrix}2\\1\end{pmatrix}$	$\begin{pmatrix}3\\2\end{pmatrix}$	end
2	$\begin{pmatrix}3\\-2\end{pmatrix}$	$\begin{pmatrix}3\\-1\end{pmatrix}$	$\begin{pmatrix}3\\0\end{pmatrix}$	end		

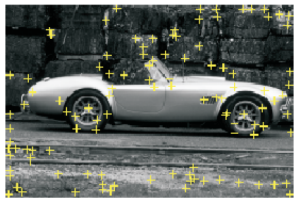
et ainsi de suite...

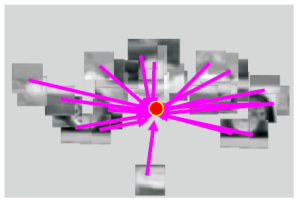
Transformées de Hough Généralisées : Variantes

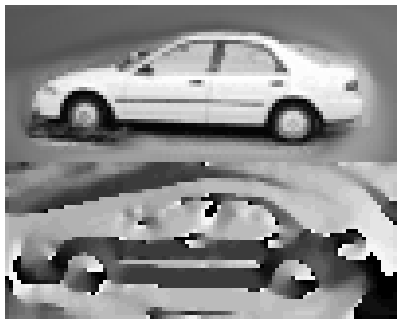
La R-Table peut être construite en utilisant différents supports, et différents indices d'apparence :



$\{n_1 : \{\delta^1_{n_1}, \delta^2_{n_1} ...\},$
$n_2 : \{\delta^1_{n_2} ...\},$
$n_3 : \{\delta^1_{n_3} ...\},$
$...\}\}$







Support :	Contour	Points d'intérêt	Tous les pixels
Index :	Normale	Dictionnaire d'imagettes	Orientation du gradient

Transformées de Hough Généralisées : Détection

Détection

Dans une transformée de Hough généralisée, chaque point, en fonction de son apparence, indique sous la forme de votes, un ensemble d'hypothèses de localisation possible du centre de l'objet. Les centres ayant reçu le plus de suffrages sont considérés comme les positions les plus probables.

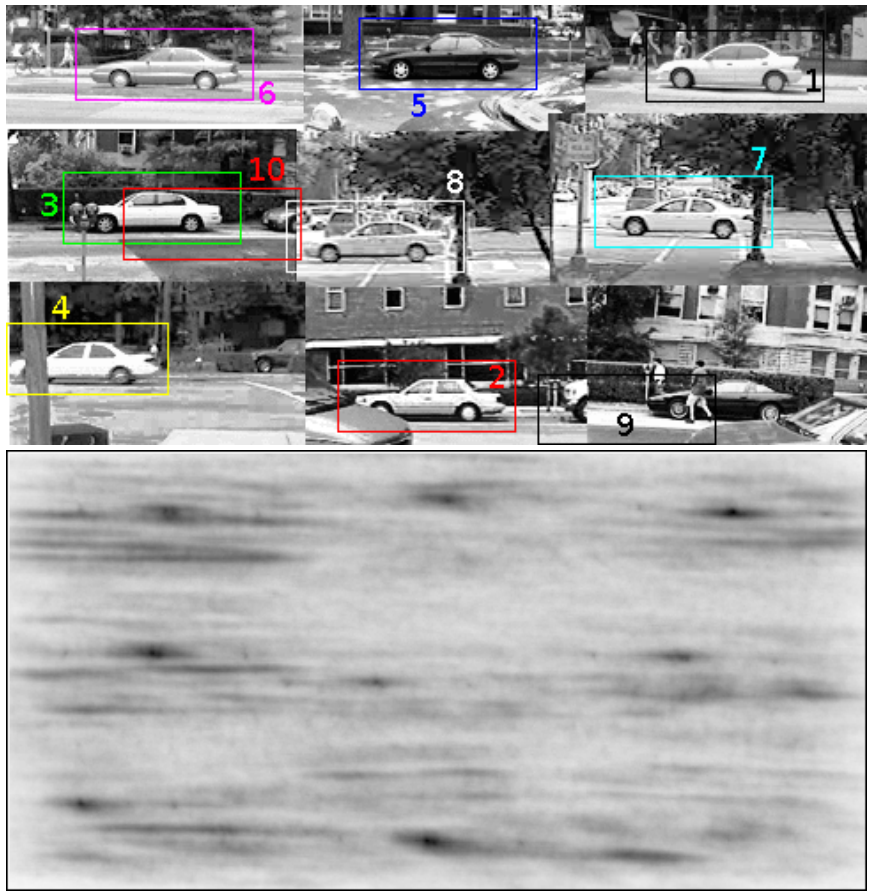
- Pour chaque point $\mathbf{x}$ de l'image on calcule son index d'apparence $\lambda(\mathbf{x})$.
- Pour tous les points $\mathbf{t}_j$ modèles tels que $\lambda(\mathbf{t}_j) = \lambda(\mathbf{x})$, on incrémente la carte de vote $H(\mathbf{x} + \mathbf{v}(\mathbf{t}_j)) += \omega(\mathbf{t}_j)$.
- La position la plus probable de l'objet est $\arg\max_{\mathbf{x}} H(\mathbf{x})$.

Transformées de Hough Généralisées : Détection

Init : $H(\mathbf{x}) = 0$ partout.
Pour tout point $\mathbf{x}$,
soit $\lambda(\mathbf{x})$ la dérivée (quantifiée).
    Pour toute occurence j de la R-Table associée à $\lambda(\mathbf{x})$, faire :

$$H(\mathbf{x} + \delta^j_{\lambda(\mathbf{x})}) \mathrel{+}= \omega^j_{\lambda(\mathbf{x})}$$

Les meilleurs objets candidats se trouvent sur les maxima of H (A droite : Transformée de Hough et les 10 meilleurs candidats "voitures").



Introduction au Mean-Shift

Mean-Shift 1/2

Le Mean-Shift est un algorithme *itératif* d'estimation de *mode* d'une distribution. Pour chaque itération :

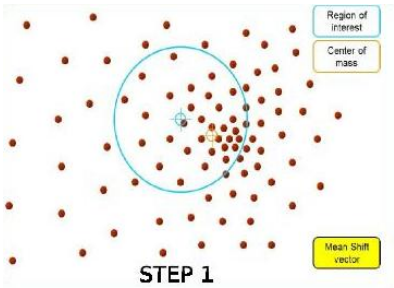
- $S = \{\mathbf{x}_i\}_i \subset \mathcal{E}$ un ensemble de points dans un espace euclidien.
- $K : \mathcal{E} \longrightarrow \mathbb{R}_+$ une fonction noyau, qui détermine la contribution de chaque point.
- $m(\mathbf{x}) = \dfrac{\sum_S K(\mathbf{x} - \mathbf{x}_i)\mathbf{x}_i}{\sum_S K(\mathbf{x} - \mathbf{x}_i)}$ la moyenne courante autour de $\mathbf{x}$.
- $v(\mathbf{x}) = m(\mathbf{x}) - \mathbf{x}$ le vecteur de mean-shift.

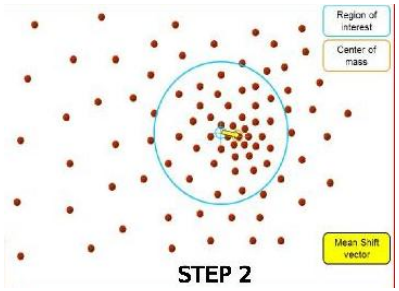
Introduction au Mean-Shift

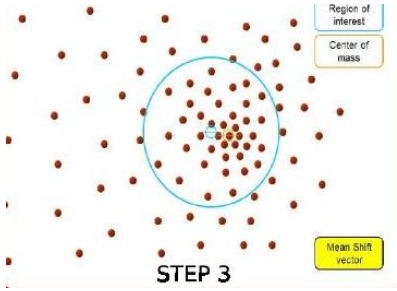
Mean-Shift 2/2

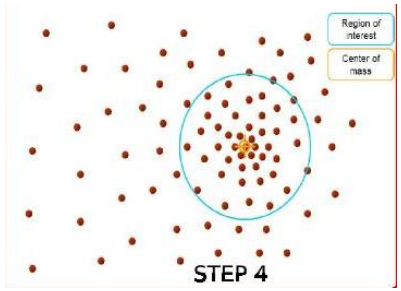
L'algorithme du Mean-Shift déplace itérativement le point $\mathbf{x}$ vers la moyenne locale. Pour chaque itération (suite) :

- $\mathbf{x} \leftarrow m(\mathbf{x})$
- L'algorithme converge lorsque $\mathbf{x} = m(\mathbf{x})$
- La trajectoire du Mean-Shift est $\{\mathbf{x}, m(\mathbf{x}), m(m(\mathbf{x})), \ldots\}$









Fonction Noyau

Le noyau K, qui détermine le poids de la contribution des différents points dans la moyenne, est en général isotrope dans l'espace euclidien, i.e.:

$$K(\mathbf{y}) = k(||\mathbf{y}||).$$

k est en général décroissante, dérivable ; on utilise souvent le noyau gaussien :

$$k(y) = e^{-y^2}, \text{ i.e. } K(\mathbf{y}) = e^{-||\mathbf{y}||^2}. \quad (1)$$

La fonction noyau K est liée à un critère de régularisation utilisé dans l'estimation d'une distribution (technique des noyaux de Parzen).

Estimation de la distribution locale

Soit f une valeur (quantifiée) de l'image en un point (Niveau de gris, couleur, ou toute caractéristique calculable en chaque pixel). L'estimation de la distribution associée à f dans le voisinage du point courant $\mathbf{x}$ défini par le support S est donnée par :

$$h_{\mathbf{x}}(u) = \frac{\sum_S k(||\mathbf{x} - \mathbf{x}_i||)\delta_u^{f(\mathbf{x}_i)}}{\sum_S k(||\mathbf{x} - \mathbf{x}_i||} \quad (2)$$

avec $\delta_u^v = 1$ si $u = v$ et $\delta_u^v = 0$ si $u \neq v$.

Similarité et index de Batthacharyya

Si $h_R(u)$ correspond à une distribution de référence (modèle), sa similarité avec la distribution locale courante peut être estimée par l'index de Batthacharyya :

$$\mathcal{B}_{\mathbf{x}} = \sum_u \sqrt{h_{\mathbf{x}}(u) h_R(u)}.$$

Le principe du Mean-Shift consiste à utiliser des distributions *lisses* pour rendre la fonction de similarité suffisamment régulière pour prédire son comportement à partir de ses dérivées spatiales (Formule de Taylor).

Similarité et index de Batthacharyya

Une approximation au premier ordre de l'index de Batthacharyya autour d'un premier point d'estimation $\mathbf{x}_0$ fournit l'approximation suivante :

$$\begin{aligned}\mathcal{B}_{\mathbf{x}} &\simeq \frac{1}{2}\sum_u \sqrt{h_{\mathbf{x}_0}(u)h_R(u)} + \frac{1}{2}\sum_u h_{\mathbf{x}}(u)\sqrt{\frac{h_R(u)}{h_{\mathbf{x}_0}(u)}} \\ &\simeq \frac{1}{2}\mathcal{B}_{\mathbf{x}_0} + \frac{1}{2}\frac{\sum_S \omega_i k(||\mathbf{x}-\mathbf{x}_i||)}{\sum_S k(||\mathbf{x}-\mathbf{x}_i||)}.\end{aligned}$$

Le 1er terme étant indépendant de $\mathbf{x}$, maximiser $\mathcal{B}_{\mathbf{x}}$ revient à maximiser le 2d terme, avec :

$$\omega_i = \sum_u \sqrt{\frac{h_R(u)}{h_{\mathbf{x}_0}(u)}}\delta_u^{f(\mathbf{x}_i)} = \sqrt{\frac{h_R(f(\mathbf{x}_i))}{h_{\mathbf{x}_0}(f(\mathbf{x}_i))}} \qquad (3)$$

Algorithme de suivi par Mean-Shift

Mean-Shift Tracking

entrée : $\{\mathbf{y}, h_R(u)\}$ (trame précédente).

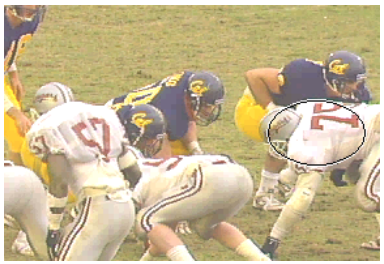
1. Calculer l'histogramme $h_{\mathbf{y}}(u)$ dans l'image courante (Eq. 2).
2. Calculer les poids ω_i en chaque point du support (Eq. 3).
3. mean-shift : Calculer la nouvelle position $\mathbf{x}$:

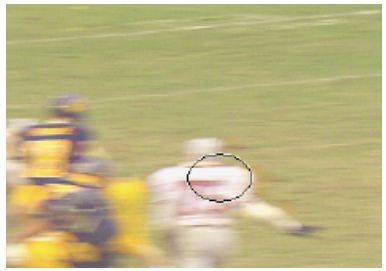
$$\mathbf{x} = \frac{\sum_S \omega_i g(||\mathbf{y} - \mathbf{x}_i||)\mathbf{x}_i}{\sum_S \omega_i g(||\mathbf{y} - \mathbf{x}_i||)} \quad (4)$$

   avec $g(x) = -k'(x)$ (Eq. 1).
4. Si $||\mathbf{x} - \mathbf{y}|| < \varepsilon$, stop. Sinon $\mathbf{y} \leftarrow \mathbf{x}$, et retour à l'étape (1).

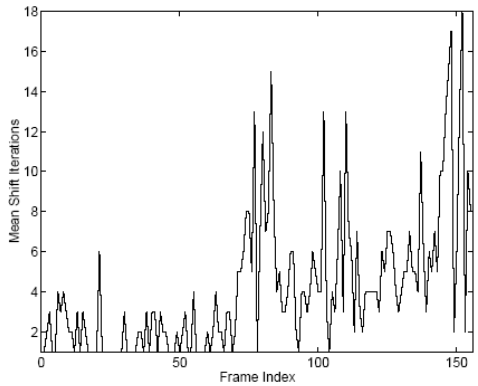
sortie : Position finale $\mathbf{x}$ et un nouvel histogramme (modèle) de référence $h_R(u) = h_{\mathbf{x}}(u)$.

Suivi par Mean-Shift



- Support : intérieur d'une ellipse.
- Espace de valeurs : RGB quantifié $16 \times 16 \times 16$.
- Nombre moyen d'itérations du Mean-Shift : 4 (Voir ci-contre).
- Travaux de Comaniciu, Ramesh et Meer **[Comaniciu03]**.



itérations / trame.

Filtrage prédictif

Filtre de Kalman

Technique d'estimation d'état constituée d'une étape de prédiction et d'une étape de correction basées sur une équation stochastique linéaire, optimale sous l'hypothèse d'une distribution gaussienne des points dans l'espace d'état.

Filtre particulaire

Technique d'estimation d'état fondée sur un échantillonnage de la distribution de l'espace d'état à partir des observations visuelles, et une propagation itérative de nouveaux échantillons extraits des images successives.

Filtre de Kalman

Le principe du filtre de Kalman est d'estimer l'état $\Theta \in \mathbb{R}^n$ du processus temporel discret gouverné par l'équation linéaire stochastique :

$$\Theta_t = A\Theta_{t-1} + BU_t + W_{t-1}$$

en utilisant une mesure $X_t \in \mathbb{R}^m$, supposée liée à Θ comme suit :

$$X_t = H\Theta_t + V_t$$

A matrice de transition $n \times n$ qui définit les liens entre deux états consécutifs.

B matrice de contrôle $n \times l$ (opt.) liée à un signal de contrôle $U \in \mathbb{R}^l$.

H matrice d'observation $m \times n$ qui associe l'état à la mesure.

V et W sont des vecteurs aléatoires indépendants, gaussiens et centrés :

$$p(V) \simeq \mathcal{N}(O, R); p(W) \simeq \mathcal{N}(O, Q)$$

Filtrage de Kalman, algorithme - d'après [Welsh01]

Init. Début avec des estimations initiales de $\hat{\Theta}_0$ et P_0.
Pour $t > 0$:

Filtre de Kalman (1) Prédiction

1. Projection de l'état courant
$$\hat{\Theta}_t^- = A\hat{\Theta}_{t-1} + BU_t$$
2. Projection de l'erreur de covariance
$$P_t^- = AP_{t-1}A^T + Q$$

Filtre de Kalman (2) Correction

1. Calcul du gain de Kalman
$$K_t = P_t^- H^T (HP_t^- H^T + R)^{-1}$$
2. Mis à jour de l'estimation avec la nouvelle mesure
$$\hat{\Theta}_t = \hat{\Theta}_t^- + K_t(X_t - H\hat{\Theta}_t^-)$$
3. Mis à jour de l'erreur de covariance
$$P_t = (I - K_t H)P_t^-$$

Filtrage de Kalman, Implantations

Modèle d'ordre 0

- Etat $\Theta_t = (x_t, y_t)$, Observation $X_t = (x_t, y_t)$.
- Matrice de transition $A = I_2$; Matrice d'observation $H = I_2$.

Modèle d'ordre 1

- Etat $\Theta_t = (x_t, y_t, v_t^x, v_t^y)$, Observation $X_t = (x_t, y_t)$.
- Matrice de transition $A = \begin{pmatrix} 1 & 0 & \delta_t & 0 \\ 0 & 1 & 0 & \delta_t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$.
- Matrice d'observation $H = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$.

Filtrage particulaire

Algorithme *Condensation* [Isard98]

Répéter :

entrée : $\{s_{t-1}^{(n)}, \pi_{t-1}^{(n)}, c_{t-1}^{(n)}\}_{n \leq N}$ l'ensemble de N anciens échantillons

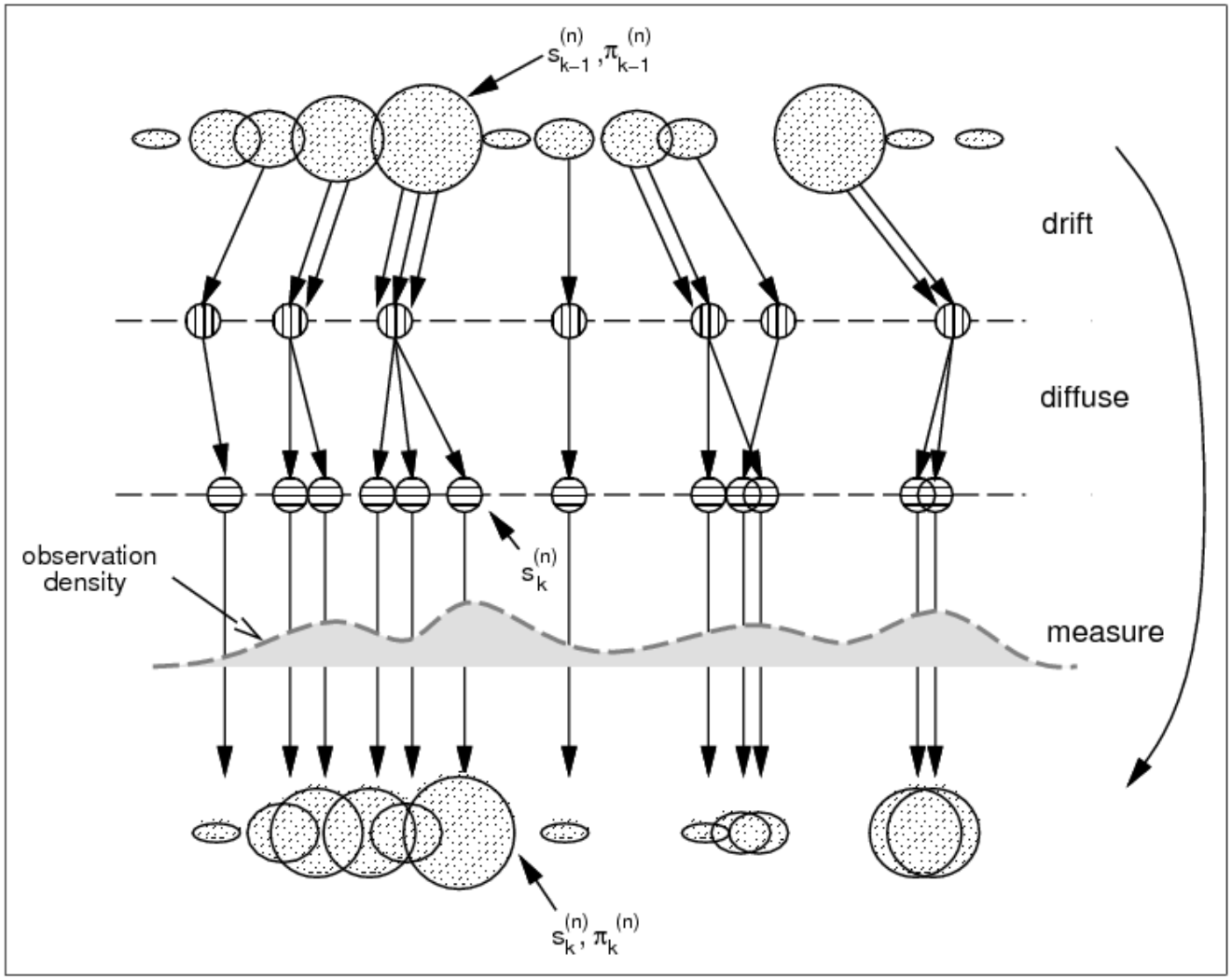
sortie : $\{s_t^{(n)}, \pi_t^{(n)}, c_t^{(n)}\}_{n \leq N}$ l'ensemble de N nouveaux échantillons

1. **Choisir** un échantillon $\hat{s}_t^{(n)}$ comme suit :
   - tirer (loi uniforme) un nombre aléatoire $r \in [0, 1]$
   - trouver le plus petit j tel que $c_{t-1}^{(j)} \leq r$
   - poser $\hat{s}_t^{(n)} = s_{t-1}^{(j)}$
2. **Prédir** le nouvel échantillon $s_t^{(n)}$:
$$s_t^{(n)} = \arg\max P(\hat{\Theta}_t = s / \Theta_{t-1} = \hat{s}_t^{(n)})$$
3. **Corriger** en fonction de la mesure et mettre à jour le poids courant :
$\pi_t^{(n)} = P(X_t / \Theta_t = s_t^{(n)})$, en normalisant pour que $\sum_{i \leq N} \pi_t^{(i)} = 1$

   puis recalculer la distribution cumulée :
   $c_t^{(0)} = 0 ; c_t^{(n)} = c_t^{(n-1)} + \pi_t^{(n)} (1 \leq n \leq N)$

Algorithme *Condensation*, suite

d'après [Isard98]



Suivi - Conclusion

Prédiction

- Permet d'estimer un état (position, taille,...) initial.
- Modèle dynamique : vraisemblance, cinématique,...
- Peut permettre de fixer un compromis entre innovation (données) et prédiction (modèle).

Observation

- Modèle d'apparence : couleur, caractéristiques...
- Description globale : vecteur, distribution, co-occurence,...
- Discrimination *vs* Invariance.
- Permet d'évaluer les états possibles autour de l'état initial.

Bibliographie - Suivi

[Jalal12] A.S. JALAL and V. SINGH
The State-of-the-Art in Visual Object Tracking
Informatica 36 (2012) 227-248

[VOT14] M. KRISTAN et al.
The Visual Object Tracking VOT2014 challenge results
Visual Object Tracking Workshop 2014 at ECCV2014, 2014

[Comaniciu03] D COMANICIU, V. RAMESH and P.MEER
Kernel-based object tracking
Pattern Analysis and Machine Intelligence, 25(5), 564-575 (2003)

Bibliographie - Filtrage prédictif

[Isard98] M. ISARD and M. BLAKE
CONDENSATION - CONditional DENSity propagATION for visual tracking
Int. Journal of Computer Vision (1998) 29(1), 5-28 (1998)

[Welsh01] G. WELSH and G. BISHOP
An Introduction to the Kalman Filter
Tutorial of ACM SIGGRAPH (2001)