Segmentation de séquences vidéo en temps réel

G. Koepfler

Introduction

But :
Détecter des objets en mouvement dans une séquence d'images.

Motivations :

- Extraire l'information des vidéos :
  → Détecter/extraire/éditer. . . les objets en mouvement.
- Première étape vers des détections de plus "haut niveau" :
  - compter les personnes,
  - déterminer le mouvement,
  - détecter des événements.
- La position des objets dans l'image permet d'approcher leur position dans l'espace.

Approches existantes

- Deux groupes :
  - *méthodes temps réel*,
    traitement en ligne, image par image ;
  - *méthodes (plus) lentes*,
    permettant l'utilisation de parties entières d'une séquence.

- Large spectre de techniques en fonction des objectifs et contraintes.

- Nécessité d'évaluer les résultats : précision temporelle et spatiale de la segmentation et du suivi, détection d'événements, . . .

Approche présentée

Simplification :

- caméra fixe (*e.g.* vidéosurveillance).

Contraintes :

- temps réel ;
- traitement en ligne ;
- peu de paramètres.

Principales difficultés :
Les changements dus aux bruits (ombres, reflets, etc) ne doivent pas être détectés comme objets en mouvement.

Exemple :
Film webcam.

Modélisation du mouvement

- Différence entre deux images successives :
  $$|I(x,t) - I(x,t+1)|.$$
- Flot optique, champ de vecteurs v tel que :
  $$I(x,t) \approx I(x+v,t+1).$$
- Comparaison à un fond donné :
  $$|I(x,t) - B(x)|.$$
  - $\oplus$ Pas de fréquence temporelle minimale,
    on détecte aussi des objets qui bougent lentement ;
  - $\ominus$ Estimation robuste du fond.

Modélisation des objets

- Méthodes basées pixels,
  les pixels sont classés statiques/fond ou dynamiques/objet.
  Exemples : algorithme W4 **(film)**, modèle gaussien et mélange de gaussiennes.
- Méthodes région,
  on part d'une région initiale que l'on adapte en minimisant une énergie.
  Contours actifs, minimisation par graph-cuts ou level sets.

Plan

Données Une image I et le fond B,

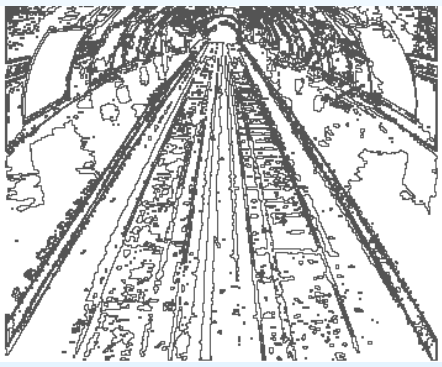
Étape 1 : Image de différences.

Étape 2 : Régions significatives

Image de différences

Principe :

- déterminer où le fond B et l'image courante I diffèrent, ceci en étant robuste aux changements de luminosité ;
- on utilise les *lignes de niveau* :
  les lignes de niveau représentent la structure dans les images : les bords des objets correspondent aux lignes de niveau avec un contraste suffisant,

;
- si un objet occulte le fond, alors il va rompre les lignes de niveau ;

Seuils

En chaque point x, on calcule

$$n_I(x) = |\nabla I|(x), \qquad n_B(x) = |\nabla B|(x),$$

$$\text{angle}(x) = \text{angle}(\nabla I(x), \nabla B(x)).$$

Trois seuils :

- e, si $n_I(x) \geq e$ on a **existence** locale de structure ;
- E, si $n_B(x) < e$ et $n_I(x) \geq E$
  on a **apparition** d'une structure, $E > e$ garantit la stabilité ;
- ϕ, si $\text{angle}(x) \geq \phi$ on a un **changement** de la structure locale.

Calcul de l'image de différences symétrique

- s'il y a changement au pixel x, on pose $D(x) =$ **blanc** :

$$[n_I(x) \geq e \text{ et } n_B(x) \geq e \text{ et } \text{angle}(x) \geq \phi]$$

$$\text{ou } [n_I(x) \geq E \text{ et } n_B(x) < e] \text{ ou } [n_B(x) \geq E \text{ et } n_I(x) < e]$$

Changement de ligne de niveau ou
"apparition / disparition" d'une ligne de niveau contrastée

- s'il n'y a pas de changement, on a $D(x) =$ **noir** :

$$[n_I(x) \geq e \text{ et } n_B(x) \geq e \text{ et } \text{angle}(x) < \phi]$$

Exemple d'images de différences

Les points blancs sont placés sur les objets en mouvement, mais sont aussi répartis un peu partout dans l'image.
En variant le paramètre ϕ :

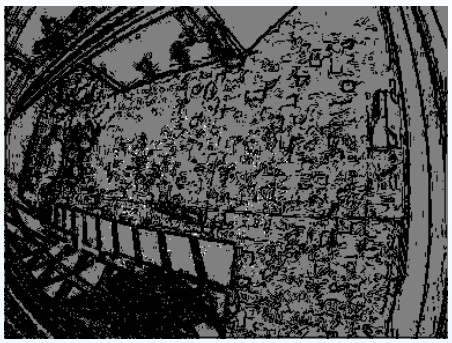
Commentaires

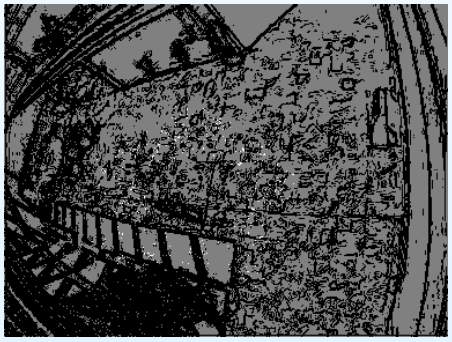
- bien que l'on va utiliser une méthode *"a contrario"* pour détecter des groupements de pixels blancs, on fixe les paramètres afin de diminuer la complexité de la méthode ;
- la valeur de E influence le comportement en cas de changements de contraste : Si $\Omega = \bigcup_i \overline{\Omega_i}$ et

$$\forall \Omega_i, \exists \alpha_i > 0, \exists \beta_i > 0, \forall x \in \Omega_i \ : \ I(x) = \alpha_i B(x) + \beta_i$$

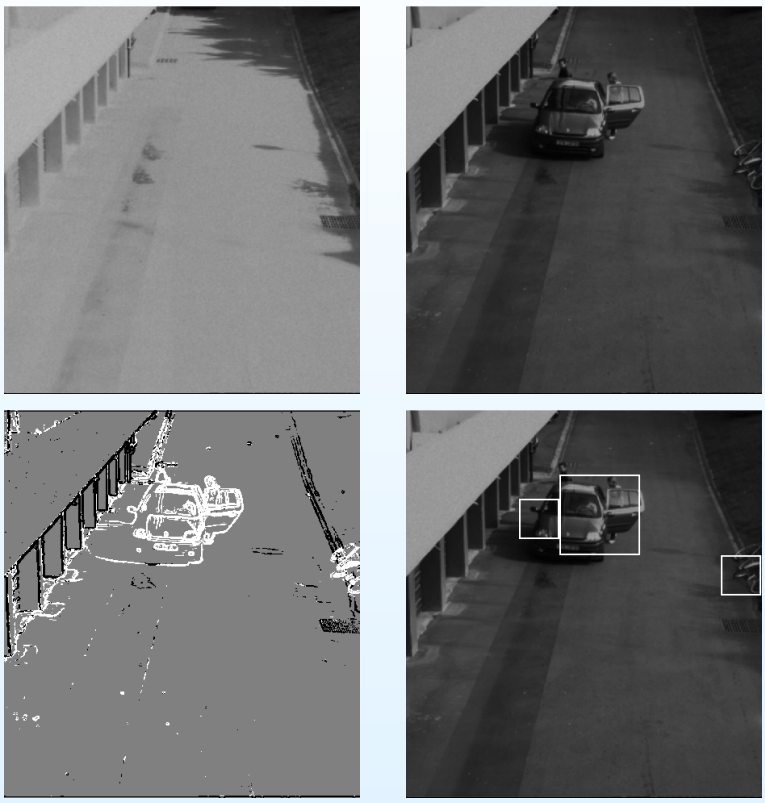
  alors si $\frac{e}{E} \leq \alpha_i \leq \frac{E}{e}$: $\forall x \in \Omega_i$, $D(x)$= **pixel noir** ou **pixel gris** ;
- des ombres ne perturbent pas l'image de différences ;
- des objets présents dans B et disparus dans I sont détectés, ce sont des fantômes :
  il faut rendre non symétrique le calcul de D ;
  la disparition de structure $[n_B(x) \geq E$ et $n_I(x) < e]$ donnera des **pixels gris**.

Ombres

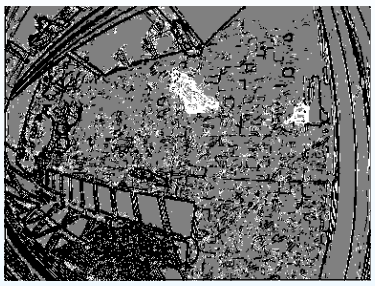


Ombres

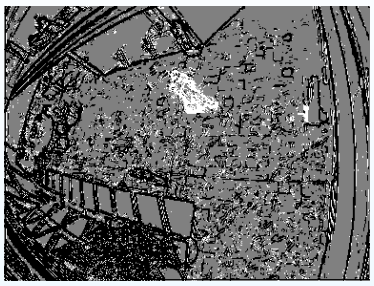


Fantômes

B



D sym.



D non sym.

I

seg. sym.

seg. non sym.

Segmentation par détection de groupements

Objet en mouvement
= groupements de pixels blancs parmi les pixels observables.

Il faut donc détecter et localiser les groupements.

Pour cela on utilise une *détection a contrario*.

Régions significatives

Détection *a contrario*

On détecte des évènements qui ne peuvent pas apparaître dans une image de bruit.

Beaucoup d'applications en images :
Détection d'alignements, de segments, de points de fuite, de contours contrastés...

Avantages de cette méthode :

- sans paramètres ;
- robuste au bruit ;
- rapidité.

$\Rightarrow$ cours A. Desolneux et J. Delon

Modèle *a priori*

On mesure la densité moyenne de pixels blancs parmi les pixels observables (blancs ou noirs) :

$$\begin{aligned} p &= P(\text{pixel blanc} | \text{ observable }) \\ &= \frac{\#\text{pixels blancs dans l'image}}{\#\text{pixels observables dans l'image}} \end{aligned}$$

Dans le modèle *a priori*, un pixel observable est blanc avec une probabilité p et noir avec une probabilité $1-p$.
Pour une région W contenant n pixels observables, la probabilité qu'elle contienne plus de k pixels blancs est donnée par la queue de la binomiale :

$$\mathbf{B}(p,n,k) = \sum_{i=k}^{n} \left(\begin{array}{c} n \\ i \end{array} \right) p^i (1-p)^{(n-i)} \ .$$

Significativité d'une région

Si on se donne un ensemble de N régions de l'image, et un seuil ϵ, on dit que W est ϵ-**significative** si

$$NFA(W) = N \cdot B(p, n, k) < \epsilon \quad \text{et si } \frac{k}{n} > p.$$

On calcule la significativité de W par :

$$S(W) = -\log(NFA(W)).$$

On considère les fenêtres comme ϵ-**significatives** si $S(W) > T$
où $T = -\log(\epsilon) > 0$

On détecte un groupement de pixels blancs dans W, si $S(W) > T$.
$\Rightarrow$ On détecte un objet en mouvement, s'il existe une fenêtre W avec $S(W) > T$.

Calcul du NFA

L'approximation de HOEFFDING donne

$$S(W) \approx n \left[p_l \log(\frac{p_l}{p}) + (1 - p_l) \log(\frac{1 - p_l}{1 - p}) \right] - \log(N).$$

avec $p_l = \frac{k}{n}$ = densité locale de pixels blancs.

Interprétation en terme de *distance de* KULLBACK-LEIBLER :
pour chaque fenêtre W, on calcule la distance entre p_l et p,
si cette distance est supérieure au seuil $T + \log(N)$, alors la fenêtre est significative.

$\Rightarrow$ On fait un seuillage adaptatif de la distance entre la densité locale et la densité globale de pixels **blancs**.

Critère de maximalité

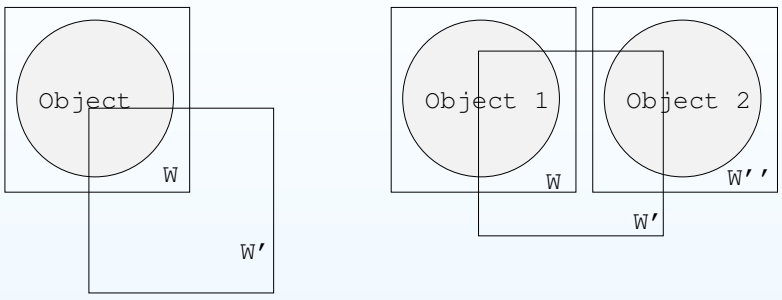
À cause de la redondance des fenêtres significatives on détermine des fenêtres maximales significatives.

Une fenêtre W est **maximale ϵ-significative** si

(a) $S(W) > T$, c.-à-d. W est ϵ-significative ;

(b) pour toute fenêtre significative W', avec $W \cap W' \neq \emptyset$,
soit $S(W) > S(W')$, donc W approxime mieux l'objet que W',
soit il existe W'', avec $W'' \cap W' \neq \emptyset$ et $S(W'') > S(W') > S(W)$,
donc W' correspond à un autre objet, localisé sur W''.

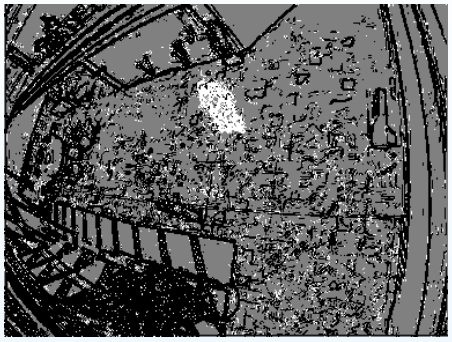
Maximalité - Algorithme

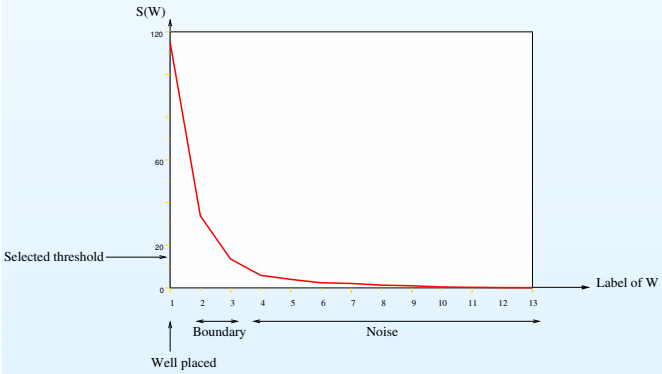


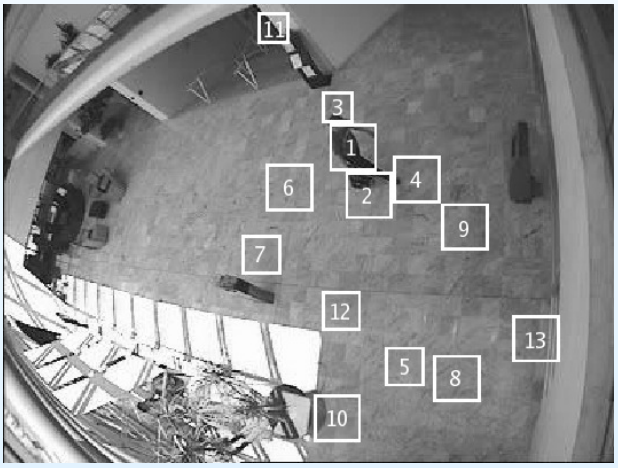
Algorithme :

1. Dans la liste initiale, trouver la fenêtre la plus significative W_{max}, l'ajouter à la liste des fenêtres maximales ϵ-significatives et l'enlever de la liste initiale ;
2. effacer toutes les fenêtres W avec $W \cap W_{max} \neq \emptyset$;
3. répéter ces deux étapes tant qu'il reste des fenêtres ϵ-significatives dans la liste initiale.

Significativité







Choix de l'ensemble de fenêtres de tests

- On considère une grille spatiale de blocs de $K \times K$ pixels.
- Les fenêtres sont des rectangles construits à partir de ces blocs.
- La position des fenêtres est aussi déterminée par la grille.

Exemple : deux tailles des rectangles 20×20 et 30×30 pixels, et une précision spatiale de taille $K = 10$.

$\Rightarrow$ les tailles des rectangles correspondent à une estimation de la taille des objets.
$\Rightarrow$ Ce choix d'ensemble de tests permet de faire un algorithme rapide.

Résultats

Les fenêtres maximales significatives donnent une segmentation approchée des objets en mouvement.

- **Séquence caviar 1.**
- **Séquence caviar 2.**
- **Séquence métro 1.**
- **Séquence métro 2.**
- **Comparaison W4 et AWD.**

Améliorer la segmentation

Amélioration de la segmentation

Une fois les régions maximales significatives détectées, on peut améliorer la résolution.

On part de l'union disjointe des fenêtres significatives, $\mathcal{O} = \cup W$, et on applique un algorithme de croissance de régions :

- on enlève les blocs $\mathcal{B} \subset \mathcal{O}$, situés aux bords de $\mathcal{O}$, si $S(\mathcal{O} \setminus \mathcal{B}) > S(\mathcal{O})$;
- on ajoute les blocs $\mathcal{B} \not\subset \mathcal{O}$, situés aux bords de $\mathcal{O}$, si $S(\mathcal{O} \cup \mathcal{B}) > S(\mathcal{O})$, et si $\mathcal{B}$ a une densité de pixels observables plus élevée que l'image de différences D ;
- répéter ces deux étapes tant qu'il reste des blocs à ajouter ou enlever.

$\rightarrow$ On utilise $S(\mathcal{O})$ comme une fonctionnelle à maximiser.

Résultats

$\Rightarrow$ Améliore la segmentation.

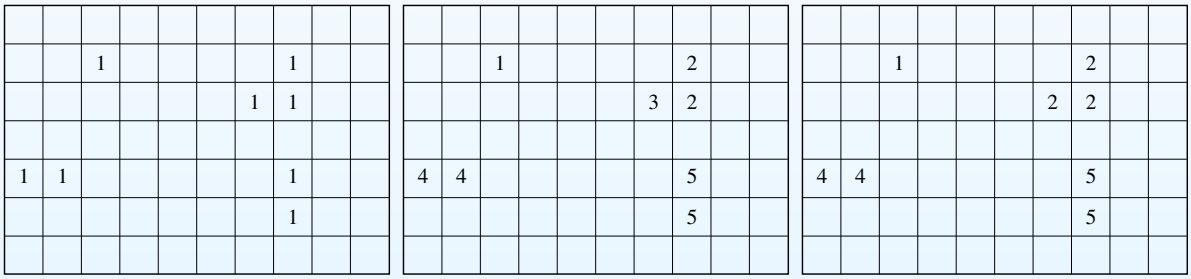
$\Rightarrow$ Critère non symétrique, sinon on ajoute à l'objet des blocs de pixels situés dans les zones homogènes, où il y a peu de pixels observables.

Séquence caviar 2.

Suivi d'objets

Suivi d'objets

A ce stade, on a ni une connectivité spatiale, deux fenêtres voisines peuvent indiquer le même objet, ni une connectivité temporelle, les images sont traitées indépendamment.
Un algorithme simple permet de numéroter les composantes connexes :



Le barycentre de chaque composante connexe permet de la relier à la région la plus proche dans l'image précédente.

Résultats suivi

- **Séquence CAVIAR.**
- **Séquence ETISEO.**
- **Séquence APRON.**

Références

Ces notes sont basées sur les documents suivants :

- S. Pelletier, PhD dissertation, *Modèle multi-couches pour l'analyse de séquences vidéo*, Université Paris Dauphine, November 2007.

- F. Dibos, G. Koepfler et S. Pelletier, *Adapted Windows Detection of Moving Objects in Video Scenes*, in SIAM Journal on Imaging Sciences, 2-1, pp. 1-19, 2009.