Analyse de la vidéo

Chapitre 3.1 - Segmentation de la vidéo

7 janvier 2015

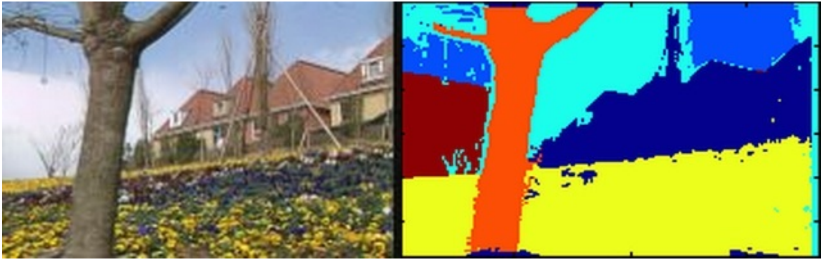
Plan de la présentation

Plan de chapitre

Définition de la segmentation

La définition formelle de la segmentation est la subdivision (partitionnement) de la vidéo en plusieurs régions R_i, $i \in \{1, \ldots, K\}$, telles que :

- $\bigcup_{i=1}^{K} R_i = \Omega$, où Ω est le domaine de la vidéo.
- $R_i \cap R_j = \emptyset$, si $i \neq j$.



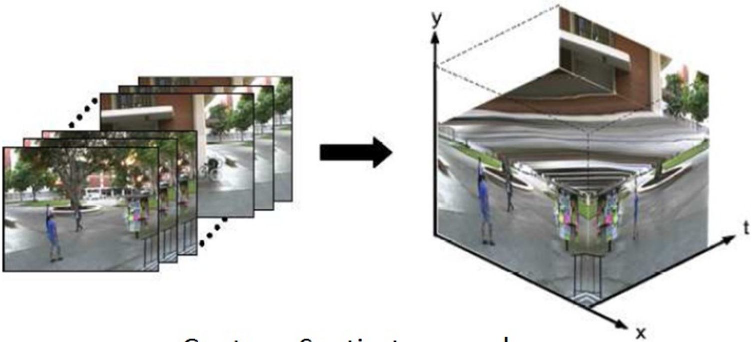
Segmentation et mouvement

La qualité de la segmentation dépend de la fiabilité de la caractéristique que l'on considère pour l'appartenance à une région donnée. Dans le cas de l'estimation du mouvement, la segmentation souffrira des même lacunes que celle-là, notamment :

- Problème d'ouverture
- Changement d'illumination
- Occlusion
- Grands mouvements

La segmentation d'effectue dans le **domaine spatio-temporel**, qui définie une séquence vidéo.

Domaine Spatio-Temporel



Contenu Spatio-temporel

Classification des familles de segmentation

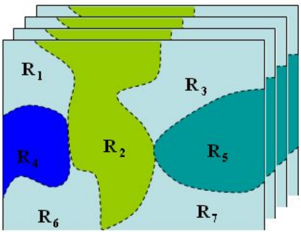
On peut classifier les différentes approches de segmentation selon les familles suivantes :

- **Segmentation spatiale** : On utilise les différents algorithmes de segmentation typiques à l'image, et ce frame par frame.
- **Segmentation temporelle** : On utilise la variation dans le temps pour segmenter le fond et les objets mobiles.
- **Segmentation sémantique** : On segmente les objets en différentes classes en utilisant l'estimation du mouvement calculée précédemment ou tout autre primitive.
- **Segmentation scénique** : On subdivise une vidéo pour fin de classification.

Plan de chapitre

Segmentation spatiale

La segmentation spatiale d'un frame en région homogène 2D est abordé en détail dans le cours *IMN 559 - Vision par ordinateur*[1].

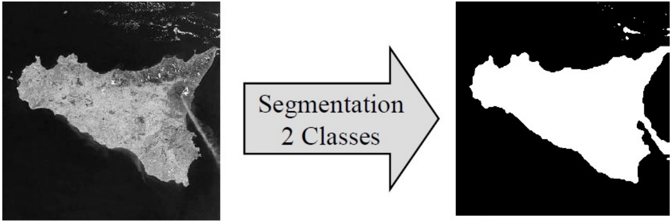


1. http://www.dmi.usherb.ca/~jodoin/cours/imn559/notes/segmentation_IMN559_2009_3pages.pdf

Segmentation spatiale

Mixture de probabilité

Le but de la segmentation spatiale est d'associer une **classe k** à un ensemble de pixel.



Tiré du cours IMN559

Segmentation spatiale

Définitions de probabilité

$P(I_t = a)$ Probabilité d'observer un pixel d'intensité (ou de vecteur de mouvement) a dans l'image I_t (*Histogramme*) ;

$P(k)$ Probabilité d'observer un pixel appartenant à la classe k ;

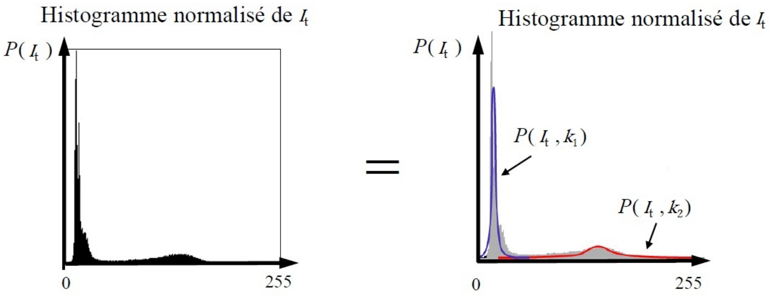
$P(I_t = a, k)$ **Probabilité jointe** d'observer un pixel d'intensité a dans l'image I_t **COMBINÉE** à la probabilité d'appartenir à la classe k ;

$P(I_t = a|k)$ **Probabilité conditionnelle** d'observer un pixel d'intensité (ou de vecteur de mouvement) a dans l'image I_t **À LA CONDITION** qu'il appartienne à la classe k (*Gaussienne*).

$$\boxed{\begin{aligned} P(I_t) &= P(I_t, k_1) + P(I_t, k_2) \\ &= \sum_{k=1}^{K} P(I_t, k) \end{aligned}}$$

Segmentation spatiale

Mixture de probabilité - Histogramme



$$\boxed{\begin{aligned} P(I_t) &= P(I_t, k_1) + P(I_t, k_2) \\ &= \sum_{k=1}^{K} P(I_t, k) \end{aligned}} \tag{1}$$

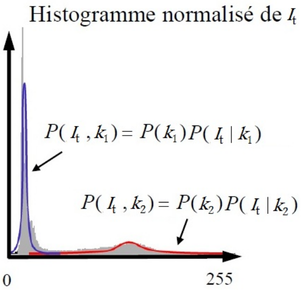
Segmentation spatiale

Mixture de probabilité - Gaussienne

Pour approximer notre distribution de caractéristiques (intensité ou mouvement), on utilise un modèle de **probabilité** comme un **noyau gaussien** :

Supposons que $P(I_t|k)$ est un noyau gaussien $G(I_t; \mu_k, \sigma_k)$, alors :

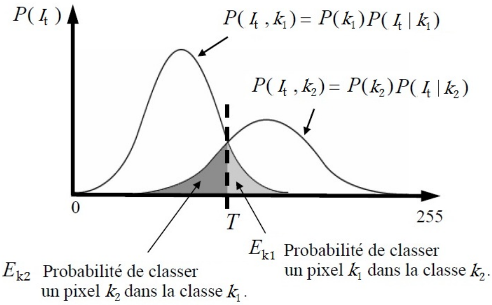
$$
\begin{aligned}
P(I_t) &= P(I_t, k_1) && + \quad P(I_t, k_2) \\
&= P(k_1) \cdot P(I_t|k_1) && + \quad P(k_2) \cdot P(I_t|k_2) \\
&= P(k_1) \cdot G(I_t; \mu_{k1}, \sigma_{k1}) && + \quad P(k_2) \cdot G(I_t; \mu_{k2}, \sigma_{k2}) \\
&= \sum_{k=1}^{K} P(k) \cdot G(I_t; \mu_k, \sigma_k)
\end{aligned}
$$



La **segmentation spatiale** consiste alors à **déterminer les paramètres des gaussiennes** G et un seuil T.

Segmentation spatiale

Mixture de gaussienne - Trouver un seuil



On trouve le seuil T qui minimise l'erreur de classification suivante :

$$\boxed{\begin{aligned} E(T) &= E_{k1}(T) + E_{k2}(T) \\ &= \int_{-\inf}^{T} P(I_t = a, k_1)da + \int_{T}^{\inf} P(I_t = a, k_2)da \end{aligned}}$$

Segmentation spatiale

Mixture de gaussienne - Déterminer le seuil

En minimisant l'Eq.3 selon T :

$$\frac{\partial E(T)}{\partial T} = 0$$
$$\cdots$$
$$\frac{P(k_1)}{\sqrt{2\pi}\sigma_{k1}} e^{-\frac{(T-\mu_{k1})^2}{2\sigma_{k1}}} = \frac{P(k_2)}{\sqrt{2\pi}\sigma_{k2}} e^{-\frac{(T-\mu_{k2})^2}{2\sigma_{k2}}} \tag{4}$$

Si $\sigma = \sigma_{k2} = \sigma_{k1}$, alors :

$$\boxed{T = \frac{\mu_{k1} + \mu_{k2}}{2} + \frac{\sigma}{\mu_{k1} - \mu_{k2}} \ln\left(\frac{P(k_1)}{P(k_2)}\right)} \tag{5}$$

Segmentation spatiale

Mixture de gaussienne - Algorithme du seuil (2 classes)

Entrées: $\boxed{I(t)}$

Sorties: $\boxed{I_s(t)}$: Image des pixels classés

Calculer T selon l'Eq.5;

pour *Tous les pixels* (x, y) *de* $I(t)$ **faire**

  si $I(x, y, t) < T$ **alors**

    $I_s(x, y, t) \leftarrow 0$

  sinon

    $I_s(x, y, t) \leftarrow 255$

retourner $I_s(t)$

Bien que cet algorithme nous permet d'estimer T, on ne détermine pas les paramètres $\mu, \sigma, P(k)$ des gaussiennes !

Plan de chapitre

Segmentation temporelle

La **segmentation temporelle** utilise l'information temporelle pour classifier les pixels des images :

Méthode de base
: La détection de changement pour un frame courant peut se faire en regardant la différence entre un frame de référence représentant le fond de la vidéo et le frame courant.

Méthode avancée
: Une famille d'algorithmes plus intelligents construit un frame de référence composé de tous les frames précédant le frame courant afin de tenir compte des variations du fond de la vidéo au cours du temps.

Segmentation temporelle

Méthode de base

Dans la segmentation temporelle, le frame de référence *Background* $B(x, y)$ est défini par la scène observée **sans objets en mouvement.**

Segmentation temporelle

Méthode de base - Camera fixe

Soient $B(x, y)$ le frame de référence *background* et $I(x, y, t)$ le frame courant. On considère, dans la méthode de base, que :

$$I(x, y, t) = B(x, y) + objets\ en\ mouvement \tag{6}$$

Suivant deux hypothèses :

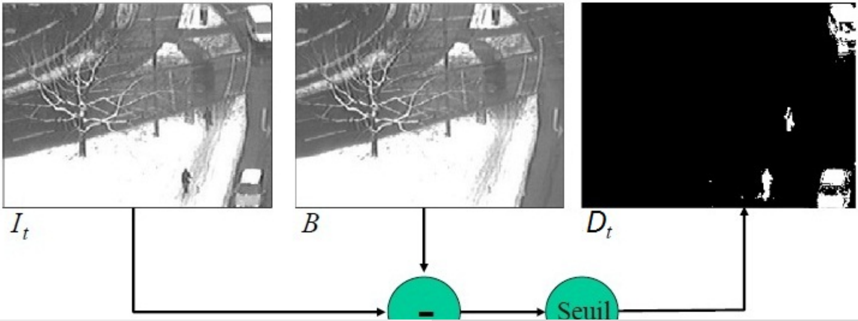
- Les objets en mouvement ont une intensité différente des pixels du fond.
- L'image de fond est toujours la même, du début à la fin.

Segmentation temporelle

Méthode de base

On peut alors supposer que la détection de mouvement est une simple soustraction de fond. dotée d'un seuillage :

$$D(x, y, t) = \begin{cases} 255 & si \quad |I(x, y, t) - B(x, y, t)| > T \\ 0 & sinon \end{cases}$$



Segmentation temporelle

Méthode de base - Camera fixe

La méthode de soustraction de fond fonctionnera bien *ssi* :

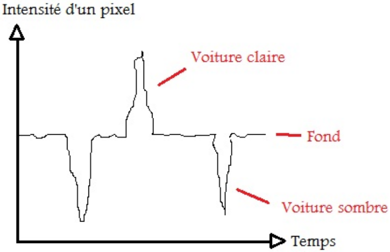
1. B est connu
2. B est constant dans le temps ;
3. I_t n'est pas une séquence « trop bruitée » ;
4. les objets du fond sont immobiles ;
5. la caméra est parfaitement fixe ;
6. les objets en mouvement ont une couleur différente du fond.

Segmentation temporelle

Méthode de base

1. Que fait-on si B est inconnu ?

On peut l'estimer à l'aide d'un ensemble d'entraînement, composée de la totalité des frames, de l'ensemble des frames précédent ou d'un seul frame.



I(t)

B

Segmentation temporelle

Méthode de base

2. Que fait-on si B varie dans le temps ?

Supposons que la scène change momentanément d'illumination ou qu'un nouvel objet arrive dans la scène pour y rester de façon stable.

On doit mettre à jour $B(t)$ à chaque temps T.

Segmentation temporelle

Méthode de base - Approche par soustraction de l'image t-1

Soient $B(x, y, t)$ le frame de référence, T un seuil prédéterminé et $I(x, y, t)$ le frame courant. La différence $D(x, y, t)$ associée au pixel (x, y) entre les instants t_r et t est calculée comme suit :

$$
\begin{aligned}
D(x, y, t) &= \begin{cases} 255 & si \quad |I(x, y, t) - B(x, y, t)| > T \\ 0 & sinon \end{cases} \\
&= \begin{cases} 255 & si \quad |I(x, y, t) - I(x, y, t-1)| > T \\ 0 & sinon \end{cases}
\end{aligned}
\tag{7}
$$

Par contre, cette méthode est très sensible au bruit et ne détectera pas de changement dans les régions uniformes.

Segmentation temporelle

Méthode de base - Approche par soustraction de l'image t-1

Une solution serait d'utiliser une fenêtre pour être moins sensible au bruit :

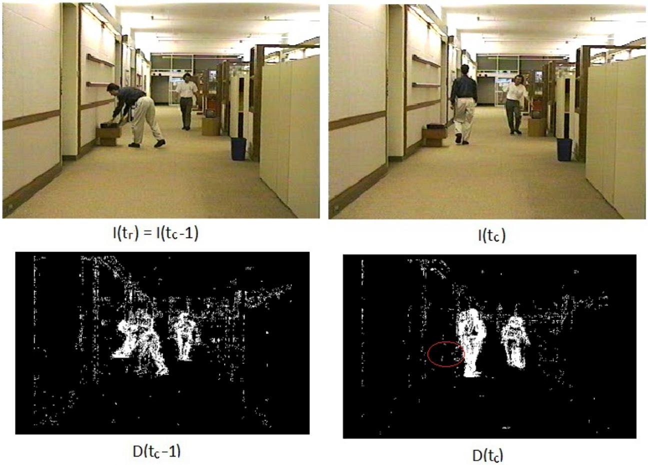
$$D(x,y,t) = \begin{cases} 255 & si \quad \frac{1}{Nb \in W} \sum_{(u,v)}^{W(x,y)} |I(u,v,t) - I(u,v,t-1)| > T \\ 0 & sinon \end{cases} \tag{8}$$

On peut aussi appliquer des filtres de *post-traitement*, que nous aborderons plus tard.

Segmentation temporelle

Méthode de base - Approche par soustraction de l'image t-1

Qu'arrive-t-il si un objet est déposé dans la scène ?



Segmentation temporelle

Méthode de mise à jour d'image de fond

3. Que fait-on si les frames I_t sont bruités ?

On souhaiterait conserver une **mémoire** de la scène afin d'identifier les objets mobiles qui deviennent **temporairement statique.**

Avec le temps, l'objet non-statique ajouté sera confondue avec le fond.

$$D(x, y, t) = \begin{cases} 255 & si \quad |I(x, y, t) - B(x, y, t)| > T \\ 0 & sinon \end{cases} \tag{9}$$

Où :

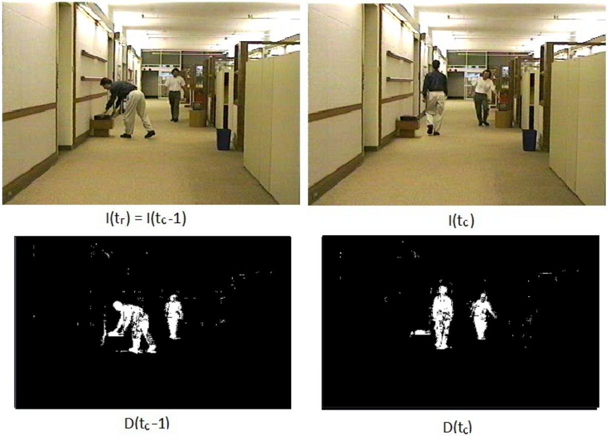
$$B(x, y, t) = (1 - \alpha) \cdot I(x, y, t - 1) + \alpha \cdot B(x, y, t - 1) \tag{10}$$

Segmentation temporelle

Méthode de mise à jour d'image de fond

Plus $\alpha \in [0, 1]$ est grand, plus la mémoire inclus les frames passés.

L'objet déposé dans la scène est maintenant détecté :



Segmentation temporelle

Algorithme de la mise à jour d'image de fond

Entrées: $\boxed{I}$: Séquence de T images

Sorties: $\boxed{D}$: Séquence d'images segmentées

$B(0) \leftarrow 0$;

$D \leftarrow 0$;

pour *t de* 1 *à* T **faire**

  pour tous les *pixels* (x, y) *de* $I(t)$ **faire**

    $B(x, y, t) \leftarrow (1 - \alpha) \cdot I(t - 1) + \alpha \cdot B(x, y, t - 1)$;

    si $|I(x, y, t) - B(x, y, t)| > T$ **alors**

      $D(x, y, t) \leftarrow 255$;

retourner D

Remarques :

- Si $\alpha = 0$, l'approche de mise-à-jour de l'image de fond est équivalente à l'approche "$t - 1$".
- Plus α est élevé, plus important sera l'**effet de traînée** sur les objet en mouvement.

Segmentation temporelle

Mixture de gaussienne

4. Que fait-on si les objets de l'arrière-plan ont du mouvement ?

Un problème avec la méthode précédente, c'est que pour bien modéliser le fond, on a besoin d'une série de frame qui n'ont pas de mouvement. Du plus, elles ne considèrent pas le bruit.

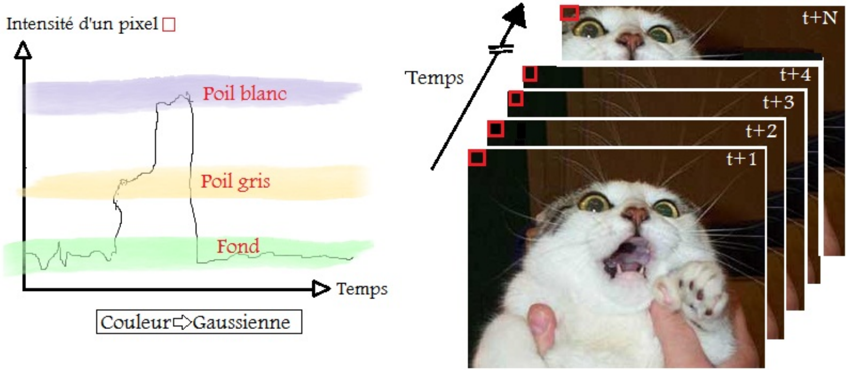
On pourrait, statistiquement, modéliser **la variation de l'*intensité* d'un pixel dans le temps.**

Nous utiliserons les **mixtures de gaussienne** pour associer les possibles valeurs du fond à une distribution gaussienne.

Segmentation temporelle

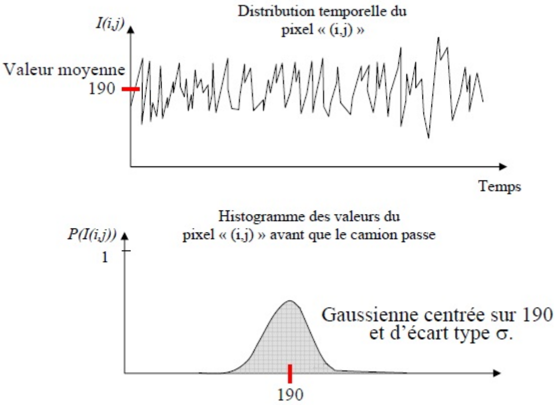
Mixture de gaussienne

En supposant que les variations d'intensités suivent une loi gaussienne :



Segmentation temporelle

Mixture de gaussienne



Tiré du cours IMN559

Segmentation temporelle

Mixture de gaussienne

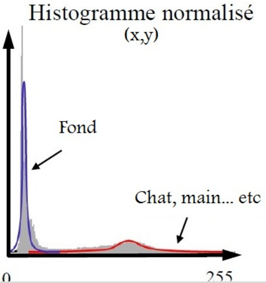
Reprenons la définition de la mixture de gaussienne (Eq.2) pour la segmentation spatiale :

$$P(I_t) = \sum_{k=1}^{K} P(k) \cdot G(I_t; \mu_k, \sigma_k)$$

Mais cette fois-ci, au lieu la distribution d'intensité de l'image, nous observerons **la distribution d'intensité d'un pixel à travers le temps** :

$$P(I(x,y)) = \sum_{k=1}^{K} P(k) \cdot G(I(x,y); \mu_k, \sigma_k) \tag{11}$$

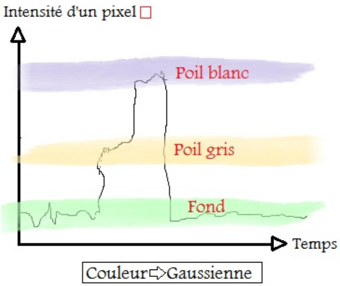
La **segmentation temporelle par mixture** consiste alors à **déterminer les paramètres des gaussiennes** G de chacun des pixels (x, y) de I et d'associé les valeurs d'intensité $I(x, y, t)$ avec ces gaussiennes.

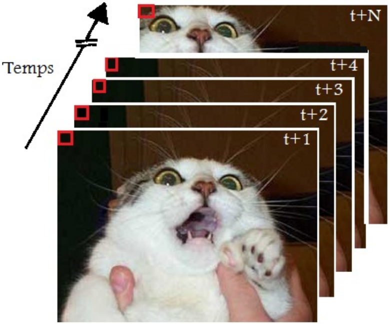


Segmentation temporelle

Mixture de gaussienne

Chaque pixel possède **sa propre mixture de gaussienne** pour décrire sa fonction de densité de probabilité (PDF).
$\rightarrow$ Approximation de l'histogramme des valeurs d'intensité du pixel dans le temps par les gaussiennes.





Segmentation temporelle

Mixture de gaussienne pondérée (Lee05)

On veut **estimer les paramètres des gaussiennes** (μ, σ, $P(k)$) en temps réel en les adaptant au fur et à mesure que l'on traite des images.

On sépare le processus de mixture de gaussienne pondérée [2] en deux étapes : un **apprentissage adaptatif** et une classification basé sur une pondération de la densité :

1. **Apprentissage adaptatif** permettant d'estimer rapidement les paramètres des gaussiennes ;
2. **Classification** basé sur la pondération de la densité des gaussiennes.

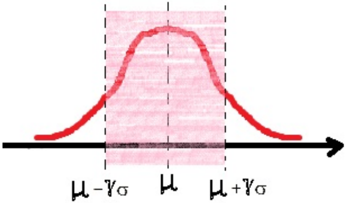
2. Lee05 - Effective Gaussian Mixture Learning for Video Background Subtraction

Segmentation temporelle

Mixture de gaussienne pondérée - Apprentissage

Soit $\gamma \simeq 3$ un seuil de variance, on veut associer l'intensité $I(x, y, t)$ avec la gaussienne la plus probable M, à la condition suivante :

$$I(x, y, t) \in \mu_k(x, y) \pm \gamma \cdot \sigma_k(x, y) \qquad (12)$$



Si plusieurs gaussiennes admettent ce critère, on doit trouver la gaussienne ayant la densité de probabilité la plus élevée :

$$\begin{aligned} P(I(x, y, t), M) &= \max_{k=1}^{K} P(I(x, y, t), k) \\ &= \max_{k=1}^{K} P_k(x, y) \cdot P(I(x, y, t)|k) \qquad (13) \\ P(I(x, y, t)|k) &= \frac{1}{\sigma_k(x,y)\sqrt{2\pi}} e^{\frac{-(I(x,y,t)-\mu_k(x,y))^2}{\sigma_k^2(x,y)}} \end{aligned}$$

Segmentation temporelle

Mixture de gaussienne pondérée - Apprentissage

Si un appariement M est trouvé

On met à jour les poids des gaussiennes en utilisant une constante d'apprentissage $\alpha \simeq 0.001$:

$$P_k(x,y) = (1-\alpha) \cdot P_k(x,y) + \alpha \cdot \left(\frac{P(I(x,y,t),M)}{\sum_{i=1}^{K} P(I(x,y,t),i)} \right) \tag{14}$$

Il faut ensuite renormaliser les poids $\sum_{k=1}^{K} P_k(x,y) = 1$.

On met à jour les paramètres de la gaussienne maximale appariée :

$$\begin{aligned} \mu_M(x,y) &= (1-\eta) \cdot \mu_M(x,y) + \eta \cdot I(x,y,t) \\ \sigma_M^2(x,y) &= (1-\eta) \cdot \sigma_M^2(x,y) + \eta \cdot (I(x,y,t) - \mu_M(x,y))^2 \end{aligned} \tag{15}$$

Où $\eta = \frac{P(I(x,y,t),M)}{\sum_{i=1}^{K} P(I(x,y),i)} \cdot \left(\frac{(1-\alpha)}{R_k(x,y)} \right) + \alpha$, et R agit comme variable de contrôle.

Segmentation temporelle

Mixture de gaussienne pondérée - Apprentissage

Si aucun appariement *M* n'est trouvé :

La gaussienne ayant la valeur de densité minimum $(P(I(x, y, t), m)$ est remplacée par une nouvelle :

- Moyenne : $\mu_m(x, y) = I(x, y, t)$.
- Écart-type : $\sigma_m(x, y) = 30$
- Probabilité de la classe : $P_m(x, y) = \alpha$
- Variable de contrôle : $R_m(x, y) = 1$

Segmentation temporelle

Mixture de gaussienne pondérée - Classification

Prenons l'expression de la distribution de densité de probabilité selon la mixture :

$$P(I(x,y,t)) = \sum_{k=1}^{K} P_k(x,y)P(I(x,y,t)|k) \tag{16}$$

On exprime la probabilité à posteriori $P(k|I(x,y,t))$ à l'aide des termes de la mixture $P_k(x,y) et P(I(x,y,t)|k)$ et une estimation de la densité de classification du *background* $B(x,y)$ $P(B(x,y)|k)$:

$$\begin{aligned} P(B(x,y)|I(x,y,t)) &= \sum_{k=1}^{K} P(B(x,y)|k)P(k|I(x,y,t)) \\ &= \frac{\sum_{k=1}^{K} P_k(x,y)P(I(x,y,t)|k)\mathbf{P}(\mathbf{B}(\mathbf{x},\mathbf{y})|\mathbf{k})}{\sum_{k=1}^{K} P_k(x,y)P(I(x,y,t)|k)} \end{aligned} \tag{17}$$

Segmentation temporelle

Mixture de gaussienne pondérée - Classification

Où $P(B(x,y)|k)$ est approximé par une fonction basée sur $P_k(x,y)$ et l'étendu σ de la gaussienne

$$P(B(x,y)|k) \simeq \frac{1}{\left(1 + e^{\frac{-96 * P_k(x,y)}{\sigma_k(x,y)} + 3}\right)} \tag{18}$$

Bref, selon une valeur $I(x,y,t)$ à évaluer, on calcule sa probabilité d'appartenir au *background* $P(B(x,y)|I(x,y,t))$ par rapport aux gaussiennes calculés lors de l'apprentissage et la densité de probabilité d'appartenance de ces gaussiennes au *background* $P(B(x,y)|k)$.

Un pixel $I(x,y)$ appartient au *background* $\Leftrightarrow P(B(x,y)|I(x,y,t)) < T$, $T \in [0.3, 0.7]$.

Algorithme de la mixture de gaussienne

Entrées: $\boxed{I, K, \alpha, \gamma}$: Séquence de T images I, K classes, constante d'apprentissage α, seuil de variance γ

Sorties: $\boxed{D}$: Séquence de T images segmentées

Initialisation **pour tous les** *Pixels* (x, y) *et Classes* $k \in [1, K]$ **faire**

  $P_k(x, y) = 0 \qquad \mu_k(x, y) = \inf \qquad \sigma_k(x, y) = 30 \qquad R_k(x, y) = 0;$

pour tous les *Images* t **faire**

  pour tous les *Pixels* (x, y) **faire**

    $\mu(x, y), \sigma(x, y), W(x, y), P(x, y) \leftarrow$ *Algorithme d'apprentissage*$(I(x, y, t), K, \alpha, \gamma)$;

    $D(x, y, y) \leftarrow$ *Algorithme de classification*$(I(x, y, t), \mu(x, y), \sigma(x, y), R(x, y), P(x, y))$;

retourner D

Algorithme mixture de gaussienne (apprentissage)

Entrées: $\boxed{I(x,y,t),\,K,\,\alpha,\,\gamma}$: Intensité $I(x,y,t)$, K classes, constante d'apprentissage α, seuil de variance γ

Sorties: $\boxed{\mu(x,y),\,\sigma(x,y),\,R(x,y),\,P(x,y)}$

pour tous les *Classes* $k \in [1,K]$ **faire**

  Calcul des $P(I(x,y),k)$ selon l'Eq.13;

si $\sum_{i=1}^{K} P(I(x,y),i) > 0$ **alors**

  pour tous les *Classes* $k \in [1,K]$ **faire**

    $q_k \leftarrow P(I(x,y),k)/\sum_{j=1}^{K} P(I(x,y),j) \qquad P_k(x,y) \leftarrow (1-\alpha)\cdot P_k(x,y) + \alpha \cdot q_k;$

  pour *La gaussienne maximale* $P(I(x,y),M)$ **faire**

    $R_M(x,y) \leftarrow R_M(x,y) + q_M \qquad \eta_M \leftarrow q_M \cdot \left(\frac{1-\alpha}{R_M(x,y)} + \alpha\right);$

    $\mu_M(x,y) = (1-n)\cdot \mu_M(x,y)(+\eta_M \cdot I(x,y,t);$

    $\sigma_M(x,y) = \sqrt{(1-n)\cdot \sigma_M(x,y) + \eta_M \cdot (I(x,y,t)-\mu_M)^2};$

sinon

  pour tous les *Classes* $k \in [1,K]$ **faire**

    $P_k(x,y) \leftarrow (1-\alpha)\cdot P_k(x,y);$

  $k \leftarrow$ Minimum des $P(x,y)$;

  $P_k(x,y) = \alpha \qquad \mu_k(x,y) = I(x,y,t) \qquad \sigma_k(x,y) = 30 \qquad R_k(x,y) = 1;$

Normaliser $P(x,y)$ ($\sum_{k=1}^{K} P_k(x,y) = 1$);

retourner $\mu(x,y),\,\sigma(x,y),\,R(x,y),\,P(x,y)$

Algorithme mixture de gaussienne (classification)

Entrées: $\boxed{I(x,y,t),\ \sigma(x,y),\ \mu(x,y),\ W(x,y),\ P(x,y)}$

Sorties: $D(x,y,t)$

num, *denom* $\leftarrow 0$;

pour tous les *Classes* $k \in [1,K]$ **faire**

$$P(k|I(x,y,t)) \leftarrow \frac{1}{\sqrt{2\pi}\sigma_k(x,y)} e^{\frac{(I(x,y,t)-\mu_k(x,y))^2}{2\sigma_k^2(x,y)}} \qquad P(B(x,y)|k) \leftarrow \left(1+e^{\frac{-96*P_k(x,y)}{\sigma_k(x,y)}+3}\right)^{-1};$$

$$num \pm P(k|I(x,y,t)) \cdot P_k(x,y) \cdot P(B(x,y)|k) \qquad denom \pm P(k|I(x,y,t)) \cdot P_k(x,y);$$

$P(B(x,y)|I(x,y,t)) \leftarrow \frac{num}{denom}$;

si $P(B(x,y)|I(x,y,t)) < (T = 0.5)$ **alors**

$D(x,y,t) \leftarrow 255$ (mouvement);

sinon

$D(x,y,t) \leftarrow 0$ (*background*);

retourner $D(x,y,t)$

Segmentation temporelle

Méthode de base - Camera mobile

5. Que fait-on si on a une caméra mobile ?

On **compense** le mouvement :

- On utilise l'estimation du mouvement global ;
- On transforme l'image selon le mouvement estimé.

On assume alors en pratique que le système mobile est alors fixe.

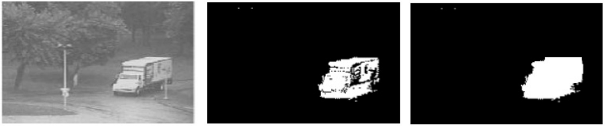
Plan de chapitre

Post-Traitement

6. Que fait-on si l'objet a partiellement les même couleurs que le fond ?

Le post-traitement permet d'améliorer les résultats de la segmentation. Elle permet entre autre de :

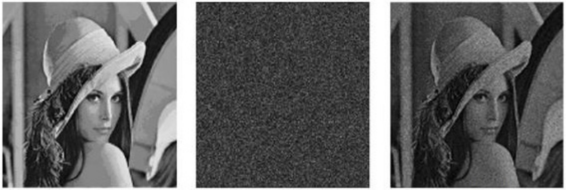
- Réduire le bruit global de l'image.
- Éliminer les faux positifs.
- Remplir les trous dans les régions segmentés (camouflage)



Post-Traitement

Filtre Median

Le filtre median permet spécialement de réduire le bruit *poivre-et-sel* et le *speckel noise*.



Lena — Speckle Noise — Lena bruitée

Une des propriétés fondamentales du filtre médian, est qu'**il ne crée pas de nouvelles valeurs de niveaux de gris dans l'image**.

→ Ce filtre est parfaitement adapté aux images binaires, puisque le résultat restera binaire.

Post-Traitement

Filtre Median

Image originale (Lenna, © Playboy).

Image avec un bruit poivre et sel affectant 25% des pixels.

Lissage linéaire 5 × 5 de l'image bruitée.

Filtrage médian 5 × 5 de l'image bruitée.

Image de: http://arthur.u-strasbg.fr/~ronse/TIDOC/FILTER/median.html

Post-Traitement

Filtre Median

Soit la fenêtre $W(x, y)$ de taille NxN, on définie le filtre médian ainsi :

$$I(x, y, t) = mediane(I(r, s, t)|(r, s) \in W(x, y)) \tag{19}$$

Comment intégrer l'information temporelle de la vidéo ?

En supposant **un déplacement petit entre les frames**, on peut concevoir un **filtre median 3D pondéré.**

Post-Traitement

Filtre Median 3D pondéré

En attribuant un filtre médian $3X3$ au frame t uni avec l'information d'un filtre médian $5X5$ au frame $t+1$, on intègre l'information du frame suivant.

Comme il y a du mouvement entre les frames, on accorde **deux fois** plus d'importance au frame actuel.

On intègre deux fois l'information du filtre median $3X3$ du frame t et une seule fois l'information du filtre median $5X5$ du frame $t+1$:

$$I(x,y,t) = mediane\left(\begin{array}{ll} I(r,s,t) | (r,s) \in W_{3\times 3}(x,y,t) & \cup \\ I(r,s,t) | (r,s) \in W_{3\times 3}(x,y,t) & \cup \\ I(m,n,t+1) | (m,n) \in W_{5\times 5}(x,y,t+1) & \end{array} \right) (20$$

Post-Traitement

Filtre morphologique

Les filtres morphologiques utilise la combinaison d'opérations mathématiques basés sur la théorie des ensembles : L'**érosion** et la **dilatation**.

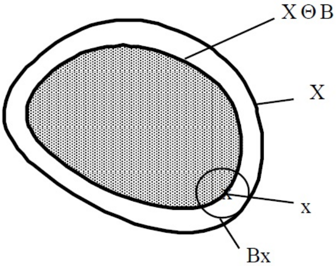
Particulièrement adapté aux image binaires, ces filtres permettent de réduire les faux-positifs (pixel blanc dans une région noire) ou les faux-négatifs (pixel noir dans une région blanche), dépendamment de la combinaison des opérations :

Post-Traitement

Filtre morphologique

♦ *Erosion* de X par B:

$$X \,\Theta\, B = \{ x \mid B_x \subseteq X \}$$



On "pèle" *X* d'une épaisseur égale à la demi-largeur de B

Répétition → la forme disparaît

IMAGE DE: http://www.limsi.fr/Individu/vezien/chapitre7.pdf

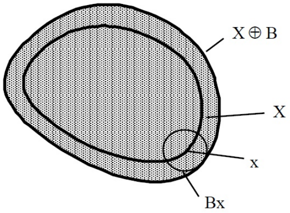
Post-Traitement

Filtre morphologique

♦ *Dilatation de X par B:*

C'est l'opération duale de l'érosion.

$$X \oplus B = \{ x \mid x = a + b,\ a \in X \text{ et } b \in B_x \}$$



• Dilater *X* revient à éroder son complémentaire

IMAGE DE: http://www.limsi.fr/Individu/vezien/chapitre7.pdf

Post-Traitement

Filtre morphologique

Soit B une fenêtre de forme quelconque. On définie respectivement l'érosion $E(t)$ et la dilatation $D(t)$ comme étant :

$$
\begin{aligned}
E(x, y, t) &= \min\,(i, j) \in B(x, y) && (21)\\
D(x, y, t) &= \max\,(i, j) \in B(x, y) && (22)
\end{aligned}
$$

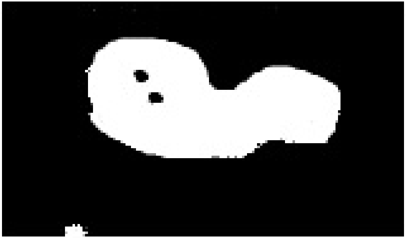
On définie l'**ouverture** O et la **fermeture** F comme étant la combinaisons des deux précédentes équations :

$$
\begin{aligned}
O(x, y, t) &= D(E(x, y, t)) && \textbf{Réduit les faux positifs} && (23)\\
F(x, y, t) &= E(D(x, y, t)) && \textbf{Réduit les faux négatifs} && (24)
\end{aligned}
$$

Post-Traitement

Filtre morphologique

Image originale



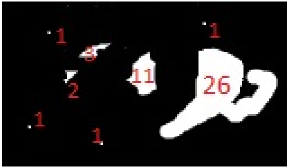
Ouverture

Post-Traitement

Flood fill - Élimination de petites régions isolés

L’algorithme du *flood fill* appliqué à la segmentation a pour but d’éliminer les petites régions isolés. Il est particulièrement utile dans le cas où un seul objet est à détecter, comme dans le *tracking*.

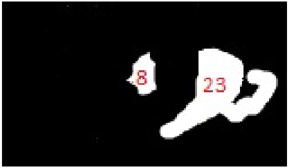
Dans le cas d’une image binaire, l’algorithme agit différemment. Imaginons que chaque région positive est représentée par une île, et que la hauteur de l’île est régie par le nombre de pixels rattachés à la région :



Post-Traitement

Flood fill - Élimination de petites régions isolés

Supposons maintenant que nous ”inondons” l’image avec du pétrole noir d’une hauteur de 3 :



On constate que les petites régions disparaissent et les grandes sont conservées.

Comment construit-on les régions ?

Post-Traitement

Flood fill - Élimination de petites régions isolés

- On parcours tous les pixels blancs de l'image binaire.
- Pour chaque pixel blanc rencontré dans les huit voisins, on garde, dans une image d'étiquette, le nombre d'occurence rencontrés.
- On recommence la vérification pour chacun de ces pixels, tout en incrémentant le nombre d'occurrence pour tous les pixels visité, jusqu'à ce qu'on ne trouve plus de voisins.
- En s'assurant de ne pas visiter les pixels déjà traités, on recommence avec un nouveau pixel blanc, tout en recommençant le décompte (nouvelle région).
- Les étiquettes forment la **carte de hauteur** des régions.