

TP de Vision Robotique

Séance 2

La séance suivante illustre quelques exemples fondamentaux d'analyse de mouvement.

On considère une séquence d'environ 20 secondes/600 images prise par une caméra installée au-dessus d'une autoroute (voir figure 1(a)). L'objectif de ce TP est de détecter les voitures en mouvement. Dans un premier temps, cela sera réalisé par la recherche du fond (dont l'aspect ne change pas). Dans un deuxième temps, on utilisera l'analyse du flot optique.

Exercice 1 — Détection du fond

1. En considérant N images converties en niveaux de gris de cette séquence (N pas trop petit mais à choisir aussi en fonction de la mémoire disponible), on peut adopter une approche très simple, en considérant pour un pixel (i, j) ses N valeurs, en calculant sa moyenne μ et son écart-type σ . Prenez une décision simple basée sur (μ, σ) pour classifier tous les pixels dans toutes les images de la séquence en fond/voiture (figure 1b). Discutez bien des choix que vous faites pour les éventuels paramètres employés.
2. En regardant la variation du niveau de gris du pixel (200,150) sur la figure 2(a), on se rend compte que le calcul de (μ, σ) est assez trompeur. Pourquoi ? On veut maintenant s'appuyer sur la cohérence temporelle d'un pixel du fond, même sur des périodes limitées. Proposez une méthode simple pour identifier des plateaux dans les niveaux de gris de pixels pour faire l'estimation de (μ, σ) plutôt en s'appuyant sur ces endroits, et essayez d'améliorer le résultat obtenu à la question précédente.
3. Faites tourner un algorithme $\Sigma\Delta$ comme celui vu en cours (un papier qui décrit plusieurs variantes se trouve ici ¹) et discutez le résultat obtenu ainsi que le choix des paramètres.
4. L'algorithme de base appliqué à la première question détecte également les ombres des voitures qui passent. Essayez, en exploitant les images couleur, d'appliquer la même idée (ou une des variantes améliorées si vous y arrivez) dans un espace de représentation plus invariant aux effets des ombres (à partir de ce qui a été vu en cours).

Observation : si vous rencontrez des problèmes avec la lecture du fichier vidéo par MATLAB en utilisant la fonction `mmreader`, vous pouvez utiliser l'utilitaire `ffmpeg` soit pour convertir le fichier vidéo dans un fichier non compressé, soit au pire pour extraire toutes les trames dans des images individuelles que vous allez charger dans une boucle en MATLAB.

1. <https://hal.science/hal-01130889/document>

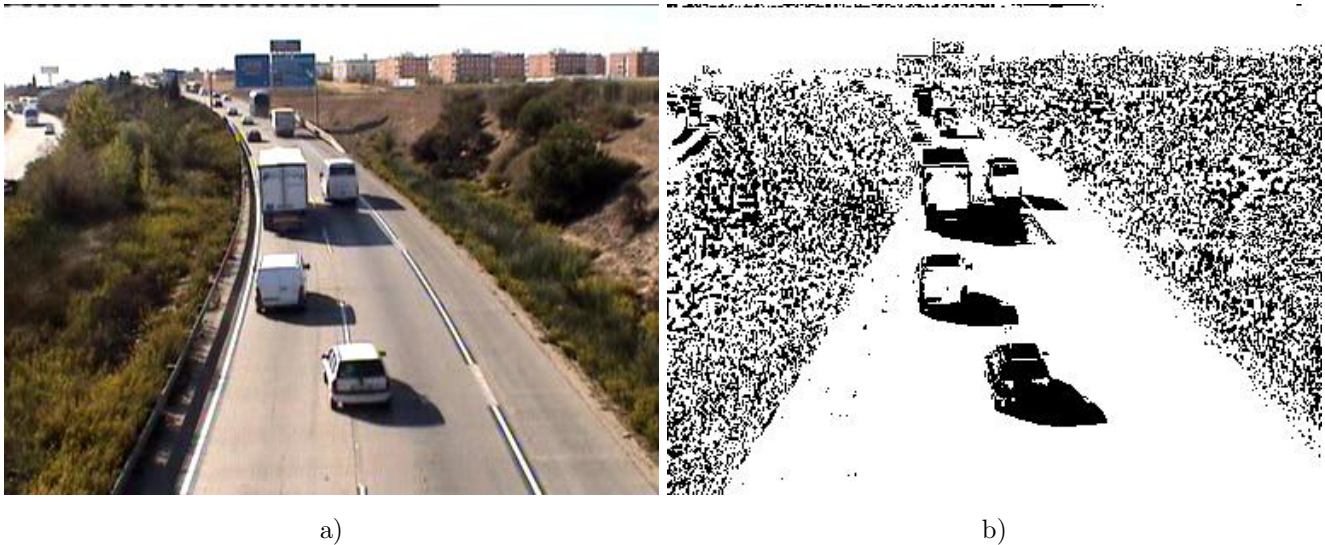


FIGURE 1 – a) Séquence vidéo initiale b) Fond en blanc.

Exercice 2 — Flot optique

Pour cet exercice on change de stratégie et on détectera les objets en mouvement.

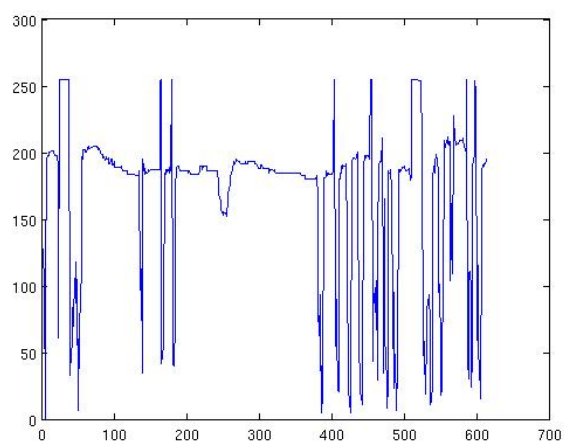
1. À partir d'un des articles référence dans ce domaine [1] disponible ici², implémentez l'algorithme proposé pour obtenir les champs de déplacement horizontaux et verticaux u et v pour une transition entre deux images. Pour visualiser les résultats on pourra utiliser la fonction `flowToColor` fournie qui associe une couleur à un pixel en fonction de la magnitude et de la direction du flot dans ce point (un exemple de résultat est sur la figure 2(b)). Le papier est très didactique (et long) mais l'algorithme est compact (et il est conseillé de faire usage au maximum de la fonction de convolution `conv2` de MATLAB). Les deux pas importants à suivre (une fois que vous avez compris la stratégie de l'algorithme) sont :

- Le calcul de E_x, E_y, E_t en utilisant les formules de la Section 7,
- L'application itérative (n fois, où n est un paramètre de la méthode) du schéma de la Section 12 qui fait intervenir également l'estimation du Laplacien de la Section 8.

Ce n'est pas mentionné dans l'article, mais appliquer un petit filtrage gaussien aux images d'entrée peut améliorer les résultats.

Observation : ne copiez pas des implémentations du web et essayez de comprendre ce qui se passe. Expliquez bien à quoi ça sert le paramètre α et testez différentes valeurs (commencez avec $\alpha = 10$). En utilisant la sortie (u, v) de l'algorithme, essayez maintenant de trouver une règle de décision simple pour décider si un pixel appartient à un objet en mouvement.

2. <http://tinyurl.com/pw8unux>



a)



b)

FIGURE 2 – a) Séquence vidéo initiale b) Visualisation du champ de flot optique.

Références

- [1] B. K. P. Horn and B. G. Schunck. Determining optical flow. *ARTIFICIAL INTELLIGENCE*, 17 :185–203, 1981.