

Cours Analyse Vidéo

TP1 : Chargement d'une vidéo avec OpenCV et Segmentation spatiale/temporelle

Note :

Ce travail doit être fait en équipe de deux ou trois (pas plus !).

Ce travail doit être fait à l'aide de la librairie opencv-python. Vous devrez vous familiariser avec la librairie, qui est utilisée pour la majorité des TPs du cours.

Le livrable à rendre est un rapport qui comprendra quelques frames de votre vidéo, l'exécution des codes fournies avec des commentaires, et des réponses aux questions posées.

Si on vous demande d'écrire un code pour une des questions, ce code devra être fourni afin de prouver que vous n'avez pas plagié.

La notation sera calculée selon la qualité de vos acquisitions, vos résultats et la clarté de votre rapport final.

I. Acquisition Chargement et Sauvegarde

1. Faites l'acquisition d'une vidéo de 5 à 8 secondes où il y a un objet en mouvement avec une caméra stable. On ne doit PAS avoir accès à une séquence où il n'y a pas d'objets en mouvement.

Si on suppose que la vidéo s'appelle « my_video.mp4 » :

- a. Chargez la vidéo à l'aide de la fonction *VideoCapture* d'opencv

```
import cv2
video = cv2.VideoCapture("my_video.mp4")
```

- b. Trouvez la fonction adéquate et affichez les informations relatives à la vidéo : hauteur, largeur, nombre de frames, et fps(frame per second)
2. Utilisez la fonction *read* pour extraire les frames de votre vidéo. Ensuite, sauvegardez-les dans le dossier de travail à l'aide de la fonction *imwrite* (ajoutez quelques frames avec numéro à votre rapport).
 3. Affichez un frame toutes les 50ms à l'aide de *imshow* (pensez à créer une boucle *while* qui s'arrête quand il n'y a plus de frames à afficher).
 4. Fermez toutes les fenêtres et libérer la lecture avec *release*

II. Segmentation spatiale: Conversion et seuillage

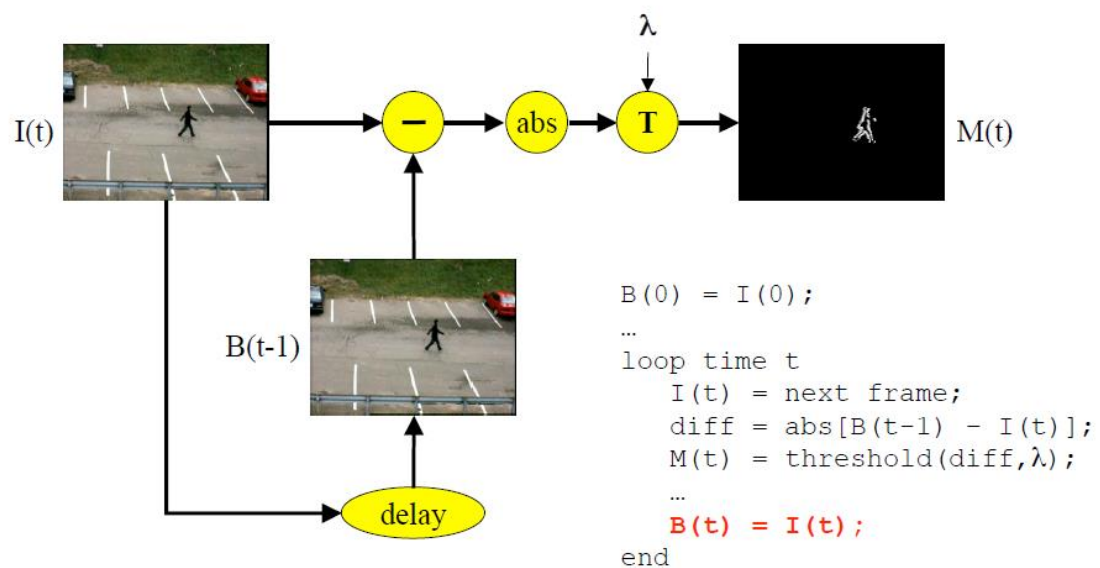
1. Reprendre avec la vidéo que vous venez de créer : On veut maintenant lire cette vidéo frame par frame et convertir ces frames de RGB aux niveaux de gris. Pour cela, vous devrez utiliser la fonction *cvtColor*. Pensez à ajouter quelques exemples à votre rapport final.

- Implantez l'algorithme de seuillage global heuristique du cours et exécutez-le sur un frame d'intérêt de votre vidéo. Pensez à choisir différent seuil T d'initialisation et observez si l'algorithme s'arrête toujours au même T_{final} ou non.
- Maintenant, on veut segmenter toutes les frames de la vidéo avec la méthode d'Otsu. Affichez dans une fenêtre la vidéo d'origine et dans une autre fenêtre la vidéo segmentée.

III. Segmentation temporelle

Reprenez avec la vidéo que vous venez de créer : On veut maintenant exploiter l'information temporelle pour détecter l'objet en mouvement dans votre séquence vidéo.

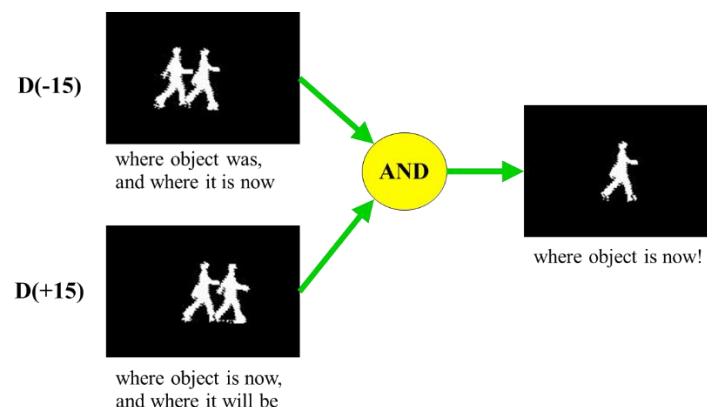
- Commencez par implanter la méthode de soustraction de l'image $t-1$: C'est la méthode de base pour estimer l'image de fond.



Affichez dans une fenêtre votre nouvelle vidéo résultante de la soustraction d'images et discutez le résultat.

Pour votre compte rendu, vous devrez mettre quelques frames de la vidéo résultante avec vos commentaires.

- Vous allez à ce stade implanter la méthode de différenciation de 3 frames



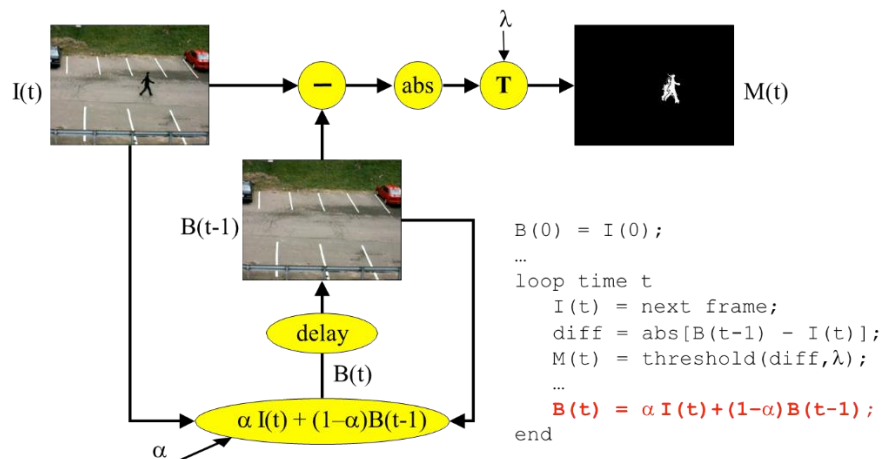
avec :

$$D(N) = \|I(t) - I(t + N)\|$$

On vous demande ici d'adapter le choix de N à votre vidéo, en fonction du frame rate de votre caméra et de la vitesse de mouvement de votre objet.

Dans votre compte rendu, vous devrez montrer des exemples de frames intéressants afin de justifier empiriquement le choix de N .

3. Maintenant, on vous demande d'implanter la méthode « Adaptive Background Substruction »



La discussion ici aura pour objectif de valider le meilleur choix de α en se basant sur vos observations.

4. Avec opencv, vous disposez de méthodes d'estimation et soustraction de fond différentes de celles vues en cours.
Pour cette dernière question, on vous demande d'identifier deux de ces méthodes, d'expliquer brièvement son fonctionnement et de montrer des résultats de leur application sur votre vidéo en y intégrant comme toujours, vos commentaires.