

Vision Par Ordinateur

TP2 : Compter des passagers

Par :
CIBAMBO Masugentwali STEVEN
&
DEGUE Akouvi MIREILLE

Université Nationale du Vietnam
Institut Francophone International / IFI
Master 2: Systèmes Intelligents et Multimédia

Janvier 2020

Contribution

- 1 Introduction
 - Context
 - Outils
- 2 Détection de mouvement
- 3 Suivi de mouvement
- 4 Comptage d'Objet en Mouvement
- 5 Conclusion

Introduction

Context

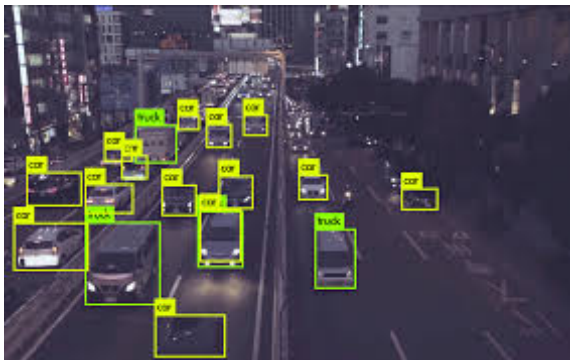


FIGURE – détection et suivi d'objets en mouvement

- surveillance, trafic routier, identification, etc.

Introduction

Outils

- ① OpenCV 3 ; pour le traitement de base d'images, détection de contours, etc
- ② Python ; pour le codage
- ③ Filtre de Kalman implémenté dans la librairie pykalman ; pour le suivi d'objets en mouvement soit la prédiction et la correction de positions mesurées

Détection de mouvement

Soustraction du Background

Il existe plusieurs techniques pour parvenir à détecter un objet en mouvement dans une séquence vidéo, dans notre cas nous avons choisi la détection par **la soustraction du background**

- ❶ **Problème** : s'il y a moindre modification du background on obtient le mauvais résultat
- ❷ **Recommandation** : utilisation de caméra fixe et éclairage stable et adéquate

Détection de mouvement

Première image de la vidéo

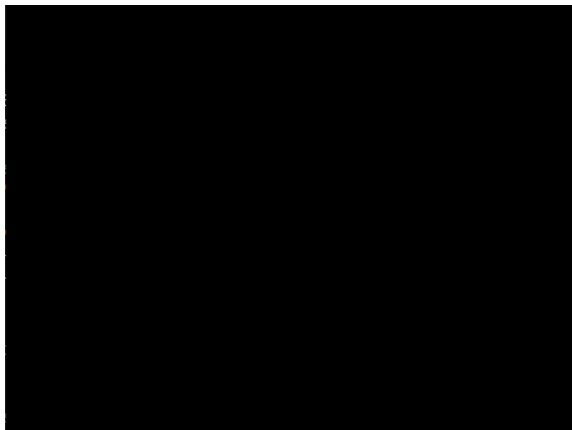


FIGURE – background

Détection de mouvement

Soustraction du background

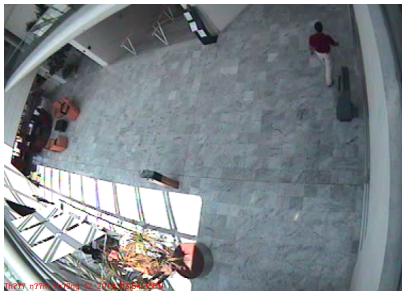


FIGURE – image originale

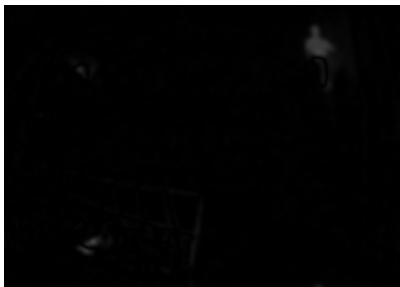


FIGURE – résultat

Détection de mouvement

Erosion et dilatation



FIGURE – résultat



FIGURE – seuillage, érode et dilate

Détection de mouvement

Objet en mouvement



FIGURE – originale



FIGURE – résultat

La taille minimum de la zone en mouvement est fixé à 500 px, pour éviter de prendre en compte même le petit changement d'éclairage

Suivi de mouvement

Filtre de Kalman

- 1 On parle qu'un objet est en mouvement quand il occupe différentes positions dans différentes images constituant une vidéo.
- 2 Souvent les positions mesurées sont rattachées des erreurs pour une raison ou une autre
- 3 Nous avons utilisé le Filtre de Kalman enfin de pouvoir prédire et corriger les positions mesurées de l'objet détecté en mouvement.

Suivi de mouvement

Initialisation du FK

```
1  from pykalman import KalmanFilter
2  from matplotlib import pyplot as plt
3  import numpy as np
4
5  MarkedMeasure = np.ma.masked_less(Measured,0)
6
7  # xinit et yinit = position ;
8  # vxinit et vyinit = vitesse
9  xinit = MarkedMeasure[0,0]
10 yinit = MarkedMeasure[0,1]
11 vxinit = MarkedMeasure[1,0]-MarkedMeasure[0,0]
12 vyinit = MarkedMeasure[1,1]-MarkedMeasure[0,1]
13 initstate = [xinit,yinit,vxinit,vyinit]
```

Suivi de mouvement

Filtre de Kalman



FIGURE – Suivi (a)

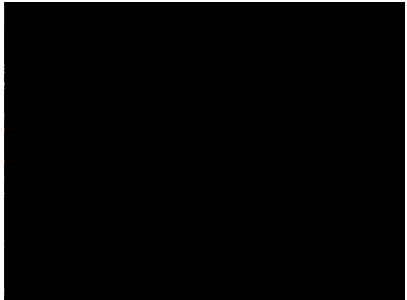


FIGURE – suivi(b)

Suivi de mouvement

Filtre de Kalman



FIGURE – Suivi (c)

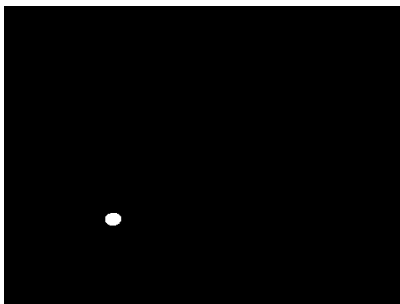


FIGURE – suivi (d)

Suivi de mouvement

Filtre de Kalman

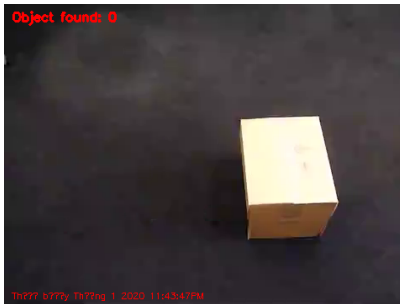


FIGURE – Suivi (e)

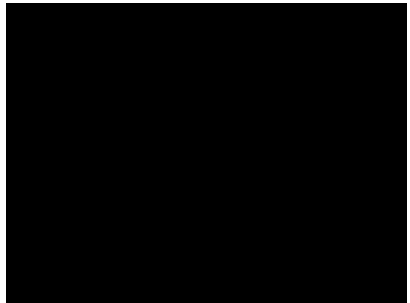


FIGURE – suivi (f)

Suivi de mouvement

Filtre de Kalman

- 1 les positions mesurées sont stockées dans un tableau ;
- 2 pour les index du tableau où aucun objet en mouvement n'est détecté ; sa valeur = -1

Suivi de mouvement

Filtre de Kalman

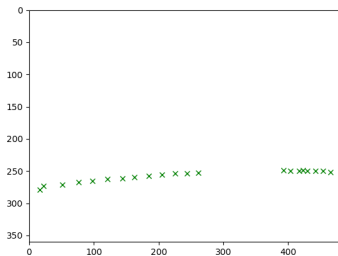


FIGURE – mesurée

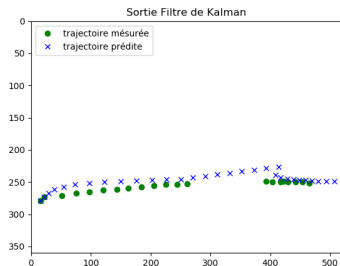


FIGURE – prédite

Suivi de mouvement

Filtre de Kalman

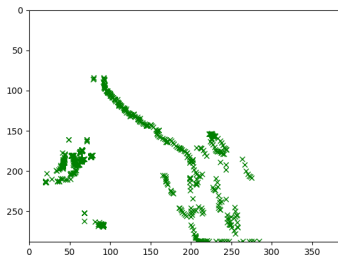


FIGURE – mesurée

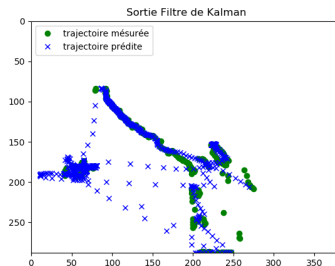


FIGURE – prédite

Comptage d'objet en mouvement

Détection de contour

- Il existe plusieurs méthodes pour compter les objets qui entrent et sortent dans une vidéo, pour notre cas nous avons utilisé celle de **la détection de contours** ;
- Et pour chaque objet objet détecté nous l'identifions pour un marqueur enfin de le distinguer des autres

Comptage d'objet en mouvement

Détetion de contour

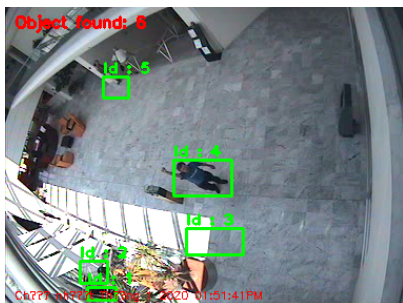


FIGURE – comptage d'objet



FIGURE – soustraction background

Conclusion

Actuellement il existe plusieurs méthode pour détecter les objets en mouvement dans une vidéo ; pour notre travail nous nous sommes servi de la méthode de **soustraction du background** ;

- 1 pour ce faire la première image de la vidéo est conservée et utilisée pour le changement susceptible pour le reste des images ;
- 2 pour la prédiction et la correction de positions nous avons utilisé le Filtre de Kalman implémenté dans la librairie pykalman de python
- 3 et enfin pour parvenir à savoir le nombre d'objets en mouvement on a utilisé la technique de la détection de contour

Références

- Computer Vision. *"Computer Vision with Python."*
Computer Vision. December 2019.
- Garima Nishad. *"Kalman Filters: A step by step implementation guide in python"* December 2019.
- Adrian Rosebrock. *"Multi-object tracking with dlib"*
December 2019.
- Code source : github