

Suivi de ballon

Réalisé par: -Abdelli Bayrem
-Missaoui Oussema
-Ayed Oussema



context

L'application de suivi de ballon a été développée dans le contexte spécifique du football (soccer) pour répondre à un défi crucial : détecter les situations de hors-jeu en temps réel. Le suivi précis de la balle est essentiel pour évaluer correctement ces situations, car la position du ballon est déterminante dans l'application de la règle du hors-jeu.



Intérêt et Motivation

-Analyse Précise des Situations de Hors-Jeu : Le suivi de ballon est au cœur de l'application, permettant une analyse fine des situations de hors-jeu. En suivant en temps réel la position du ballon, l'application contribue à déterminer avec précision si un joueur est en position de hors-jeu au moment précis du passage du ballon.

-Application de Techniques Avancées de Vision par Ordinateur : Le développement de cette application met en avant une expertise technique avancée, en particulier dans le domaine du suivi d'objets et de l'utilisation du **filtre de Kalman**. Ces techniques avancées sont nécessaires pour assurer un suivi précis du ballon dans des environnements dynamiques tels que les matchs de football.

-Modélisation de la Trajectoire Anticipée : En utilisant des modèles de prédiction , l'application pourrait anticiper la trajectoire future du ballon en fonction de sa vitesse actuelle, de son accélération, et des conditions environnementales. Cela permettrait d'anticiper les situations de hors-jeu potentielles, offrant ainsi une analyse prédictive aux utilisateurs de l'application.

Outils, Technologies et Bibliothèques Utilisés

OpenCV (Open Source Computer Vision) :

Description : OpenCV est une bibliothèque très populaire pour le traitement d'image et la vision par ordinateur. Elle fournit des outils puissants pour la détection d'objets, le suivi d'objets, le traitement d'image, etc.

Utilisation dans le Projet :

- Détection de la balle.
- Suivi des joueurs.
- Calcul de l'histogramme et de la projection arrière.
- Transformation de perspective.
- Détection de contours.

Outils, Technologies et Bibliothèques Utilisés

NumPy :

Description : NumPy est utilisé pour la manipulation de tableaux et les opérations matricielles. C'est une bibliothèque essentielle pour les calculs numériques en Python.

Utilisation dans le Projet :

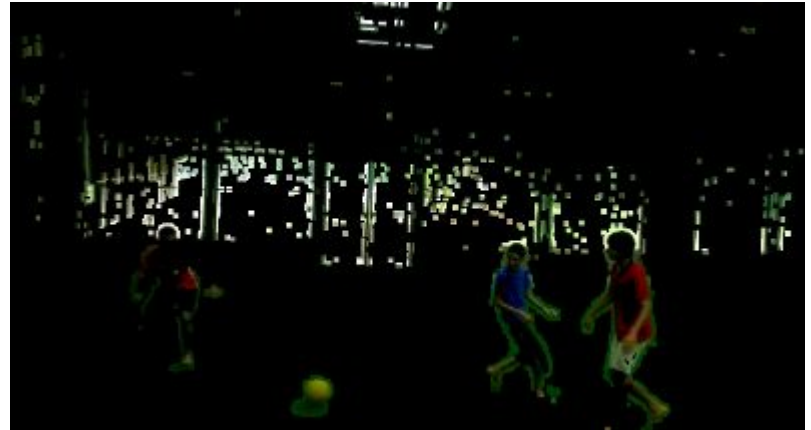
- Manipulation des coordonnées des points.
- Calculs matriciels.

Outils, Technologies et Bibliothèques Utilisés

Algorithme de Soustraction d'Arrière-plan (Background Subtraction) :

Description : L'algorithme MOG2 (Mixture of Gaussians) est utilisé pour soustraire l'arrière-plan de la vidéo, isolant ainsi les objets mobiles.

Utilisation: Pour obtenir un masque.

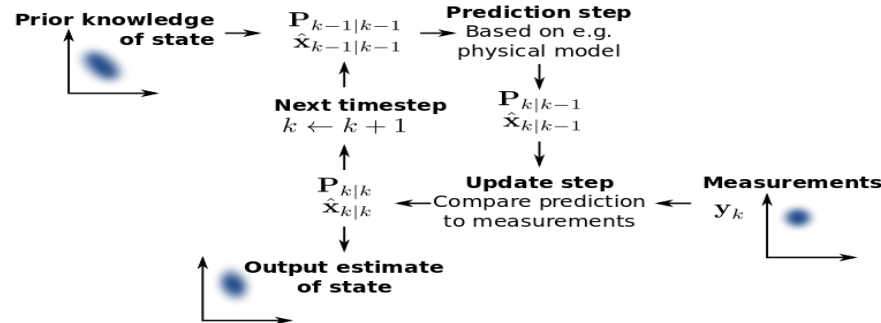


nous avons utilisé la fonction `calcBackProject`
d'OpenCV pour générer une image en niveaux de gris .



Algorithme de Kalman :

Le filtre de Kalman est un algorithme de filtrage récursif qui est souvent utilisé pour estimer l'état d'un système dynamique à partir de séries temporelles d'observations bruitées. Dans le contexte de ce projet, le filtre de Kalman est utilisé pour prédire la position future de la balle en se basant sur les observations passées.



Fonctionnement du Filtre de Kalman pour le Suivi de la Balle :

Coordonnées de la Balle :

Le filtre de Kalman est utilisé pour suivre la position 2D de la balle sur le terrain de jeu.

Prédiction de la Trajectoire :

En utilisant les observations passées, le filtre de Kalman prédit la trajectoire future probable de la balle.

Compensation du Bruit et des Erreurs :

En prenant en compte l'incertitude associée aux observations et au modèle dynamique, le filtre de Kalman atténue les effets du bruit et des erreurs, améliorant ainsi la précision du suivi.

Adaptabilité Dynamique :

Le filtre de Kalman ajuste dynamiquement ses prédictions en fonction des variations de la trajectoire de la balle, ce qui le rend adaptatif aux changements de vitesse et de direction.

Cas d'utilisation :

1. Suivi de la Balle :

L'application utilise des techniques de vision par ordinateur et de suivi d'objets pour détecter et suivre la balle de football en temps réel sur le terrain.

2. Détection des Joueurs :

Les joueurs sont détectés et suivis sur le terrain à l'aide d'algorithmes de traitement d'image. Cela permet de surveiller les mouvements des joueurs et d'analyser les interactions avec la balle.

3. Filtrage de Kalman :

Le filtre de Kalman est appliqué pour prédire la trajectoire future de la balle, améliorant la stabilité du suivi et compensant les erreurs introduites par le bruit dans les observations.

4. Détection des Actions (Passe, Offside) :

L'application analyse les mouvements de la balle et des joueurs pour détecter des actions spécifiques telles que les passes et les situations d'offside.

Réalisation

L'application combine plusieurs éléments pour atteindre les fonctionnalités décrites :

Traitement d'Image :

Utilisation de la bibliothèque OpenCV pour le traitement d'image, la détection d'objets, et l'extraction d'informations à partir du flux vidéo.

Filtre de Kalman :

Implémentation du filtre de Kalman pour prédire la trajectoire future de la balle en se basant sur les observations passées.

Détection des Joueurs :

Utilisation d'algorithmes de détection de contour pour suivre les joueurs sur le terrain.

Interface Utilisateur :

Une interface utilisateur (GUI) peut être intégrée pour fournir une visualisation en temps réel du suivi de la balle, des actions détectées, et d'autres informations pertinentes.

Approche du problème

Suivi de la Balle (Couleur + Seuil)

Extraction d'un échantillon de la balle --> Conversion dans l'espace colorimétrique HSV --> Soustraction de l'arrière-plan

Détection du contour de la balle --> Génération d'un cercle englobant minimum --> Suivi avec historique

Suivi des Joueurs (Rétroprojection d'Histogramme) :

Utilisation de la rétroprojection d'histogramme pour détecter les joueurs --> Seuillage de l'image rétroprojetée

Application d'érosion et de dilatation pour supprimer le bruit --> Obtention des contours des maillots des joueurs --> Estimation de l'emplacement des pieds

Génération de la Vue de Dessus :

Définition des extrémités du terrain par l'utilisateur --> Utilisation de la fonction `getPerspectiveTransform` pour obtenir la matrice d'homographie

Calcul des coordonnées en vue de dessus des pieds des joueurs et de la balle --> Tracé de la ligne de hors-jeu en inversant le mappage des extrémités

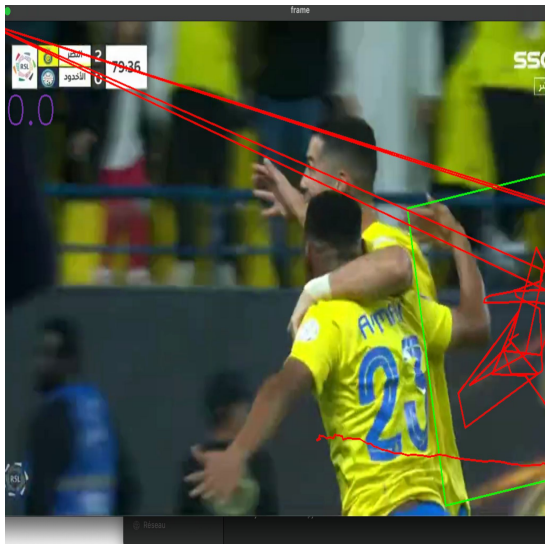
Du CAMShift au Filtre de Kalman

Au commencement de notre projet, nous avons choisi d'utiliser l'**algorithme CAMShift** pour effectuer le suivi d'objets dans nos séquences vidéo. L'algorithme CAMShift est connu pour sa capacité à suivre des objets en mouvement dans des scénarios changeants en ajustant dynamiquement la taille de la fenêtre de recherche en fonction de la distribution de couleur de l'objet.

Cependant, au fur et à mesure que nous avançons dans le développement, nous avons identifié des limitations dans l'efficacité de l'algorithme CAMShift, notamment en présence de changements brusques d'échelle, d'occlusions temporaires, ou lorsque l'objet sortait temporairement du champ de vision. Pour surmonter ces défis, nous avons décidé de migrer vers l'utilisation du **filtre de Kalman**.

Le filtre de Kalman, un outil puissant en estimation et prédiction, nous a permis d'améliorer la robustesse du suivi d'objets. Contrairement à CAMShift, le filtre de Kalman intègre un modèle de mouvement qui peut prédire la position future de l'objet, même en cas d'occlusion temporaire. De plus, le filtre de Kalman peut également traiter des données bruitées, offrant ainsi une meilleure stabilité dans des conditions environnementales difficiles.

Limites d'algorithme Camshift



CAMSHIFT peut faire face à des situations où l'apparence de l'objet change de manière significative, telles que des occultations, des variations de luminosité, ou lorsque l'objet subit des transformations brusques.

Conclusion

Lorsqu'il s'agit de suivre la trajectoire d'une balle tout au long du temps, surtout si le mouvement est relativement prévisible, un filtre Kalman pourrait être utile.

Si l'apparence de la balle peut changer de manière significative (par exemple, en raison de la rotation ou de la mise à l'échelle), et vous avez besoin d'un tracker robuste qui peut s'adapter à ces changements CAMSHIFT pourrait être plus approprié.

En conclusion, le choix entre ces algorithmes est en fait basé sur les caractéristiques spécifiques de votre problème, tels que la nature du mouvement de la balle, l'environnement et les types de variations que l'algorithme de suivi doit prendre en charge.

Utilisation du programme

-\$ python Offside_detection.py -v 'video.mp4'

-Pendant l'exécution du programme, appuyez sur « i » pour saisir et « q » pour quitter.(voir demo.mp4)

lien:

[drive:https://drive.google.com/drive/folders/1yPeUA3GiZYDWsLkHh1Ng57gAtyuVmmDN](https://drive.google.com/drive/folders/1yPeUA3GiZYDWsLkHh1Ng57gAtyuVmmDN)