Arnaud Baradat

Visual Object Tracking Report

EPITA

SCIA 2024

1. Introduction

Le suivi d'objets est une technique de vision par ordinateur qui vise à détecter et suivre les déplacements d'objets dans des séquences vidéo.

Pour ce projet, nous avons porté notre intérêt sur le suivi de personnes et nous avons utilisé la vidéo ADL-Rundle-6 disponible dans le cadre du MOT-Challenge 2017, qui propose un environnement de test standardisé pour le suivi.

L'approche adoptée combine des méthodes traditionnelles et avancées, incluant le filtre de Kalman pour le suivi d'un seul objet, l'algorithme hongrois et l'Intersection sur Union (IoU) pour le suivi multi-objets, ainsi que l'intégration de caractéristiques visuelles et de réseaux de neurones convolutionnels pour améliorer la précision et la robustesse du suivi.

2. Méthodologie

Nous dresserons un tableau des différentes étapes du projet pour avoir une idée de comment nous sommes arrivés au résultat actuel.

2.1 Suivi d'objet unique avec l'IoU

Ceci est la première implémentation fonctionnelle que nous ayons fait. Elle n'a que pour seule métrique l'IoU, elle utilise un matching naïf.

L'Intersection sur Union (IoU) est utilitśee pour évaluer la correspondance entre les objets détectés et les objets suivis.

- Calcul de l'IoU : Pour chaque paire constituée d'une détection et d'une piste existante, l'IoU est calculé en divisant l'intersection de leurs boîtes englobantes par leur union.
- Utilisation dans le Suivi : Un score IoU élevé indique une forte correspondance, suggérant que la détection et la piste représentent le même objet.

Ce score est essentiel pour associer correctement les détections aux pistes existantes, en particulier dans des scènes avec plusieurs objets en mouvement.

Combiné à un algorithme de matching naïf, nous avons notre première implémentation fonctionnelle.

2.2 IOU Tracker amélioré avec des filtres de Kalman.

Cette approche utilise toujours un matching naïf mais utilise des filtres de kalman pour avoir une meilleure prédiction des positions attendues des tracks connus dans la frame suivante.

Dans ce projet, le filtre de Kalman est utilisé pour estimer en continu la position et la vitesse d'un objet dans une séquence vidéo 2D. Il fonctionne en deux phases principales :

- Phase de Prédiction : estimation de l'état futur de l'objet (position et vitesse) en utilisant le modèle de mouvement actuel.
- Phase de mise à jour : Ajustement de l'estimation à partir des nouvelles mesures (position de l'objet détecté) pour affiner la précision du suivi.

Ainsi, nous pouvons améliorer la mise en correspondance pour des tracks existants.

2.3 IOU Tracker amélioré avec filtres de Kalman & l’algorithme de matching Hongrois

L'évolution du système de suivi vers un traqueur IOU amélioré implique l'intégration de l'algorithme hongrois, une méthode d'optimisation pour résoudre les problèmes d'assignation. Dans le contexte du suivi multi-objets, cet algorithme joue un rôle crucial pour associer de manière efficace et précise les détections d'objets aux pistes existantes.

Fonctionnement de l'Algorithme Hongrois : L'algorithme opère en utilisant la matrice de similarité, basée sur les scores IoU, comme entrée. Son objectif est de trouver l'assignation optimale qui maximise la somme globale de la similarité (ou minimise la distance) entre les détections et les pistes.

Application dans le Suivi : Après le calcul des scores IoU pour chaque paire de détection et de piste, l'algorithme hongrois est appliqué pour déterminer les meilleures correspondances. Cette approche assure que chaque détection est assignée à la piste la plus appropriée, tout en évitant les doublons et en maximisant l'efficacité globale du suivi.

Cette méthode d'optimisation contribue à une meilleure gestion des identités des personnes suivies, un aspect crucial dans le suivi multi-objets.

2.4 Multi-Object Tracking Avancé

2.4.1 Intégration d’embeddings ResNet18 & Histogrammes de couleurs

Pour affiner davantage le système de suivi, nous avons intégré des embeddings basés sur ResNet18 pour chaque boîte englobante détectée. Cette amélioration permet d'ajouter une métrique de similarité supplémentaire basée sur les caractéristiques visuelles des objets.

Les embeddings ResNet18 capturent des informations visuelles détaillées pour chaque objet, allant au-delà de leur position et taille. Ces embeddings offrent une représentation riche de l'apparence de l'objet, facilitant ainsi la distinction entre des objets aux caractéristiques physiques similaires.

En combinant les embeddings ResNet18 avec les scores IoU, le système peut effectuer des correspondances plus précises en tenant compte à la fois de la position géométrique et des attributs visuels des objets.

2.4.2 Re-calculation des détections avec YoloV7

La seconde amélioration majeure est l'adoption de YoloV7 pour la détection des objets. YoloV7, étant un des modèles de détection d'objets les plus avancés et rapides, apporte une amélioration significative en termes de précision et de vitesse de détection.

L'utilisation de YoloV7 entraîne une détection plus fiable et précise des objets, ce qui se répercute directement sur l'efficacité du suivi. La précision accrue des détections contribue à réduire les erreurs dans les phases ultérieures du processus de suivi.

En combinant les embeddings ResNet18 et le modèle de détection YoloV7, notre système de suivi multi-objets atteint une sophistication et une précision nettement supérieures, permettant un suivi plus robuste et fiable dans divers scénarios. Ces améliorations illustrent l'efficacité de l'intégration de techniques avancées de vision par ordinateur dans le suivi d'objets.

2.4.3 Résumé de l’architecture actuelle

- Nouvelles détections Yolo: Utilisation d'un détecteur Yolo pour l'identification précise des personnes.

- Métriques de Similarité: (somme pondérée)
    - IoU: Superposition géométrique entre les boîtes englobantes.
    - Similarité Cosinus: Basée sur les embeddings ResNet18.
    - Distance L2: Pour les histogrammes de couleur.
- Pré-Matching avec Filtres de Kalman: Prédiction des positions attendues des objets pour les pistes existantes.
- Matching avec l'Algorithme Hongrois: Association optimale des détections aux pistes.

2.4.4 Aparté sur les métriques

Le choix d’une similarité cosinus pour les embeddings à été fait car les vecteurs sont de grande dimensionnalité (cette métrique fonctionne bien dans ce cas) ainsi que de part sa plage de valeurs dans $[-1, 1]$ ce qui permet une grande affinité avec l’IoU.

Le choix de faire une distance L2 pour les histogrammes de couleurs était un choix plus arbitraire, venant d’une volonté d’expérimenter avec les différentes métriques que je connaissais. Il est intéressant de noter que je n’ai pas vu de grande différence entre l’utilisation d’une distance L2 normalisée et une similitude cosinus normalisée. Notez que les trois métriques sont:

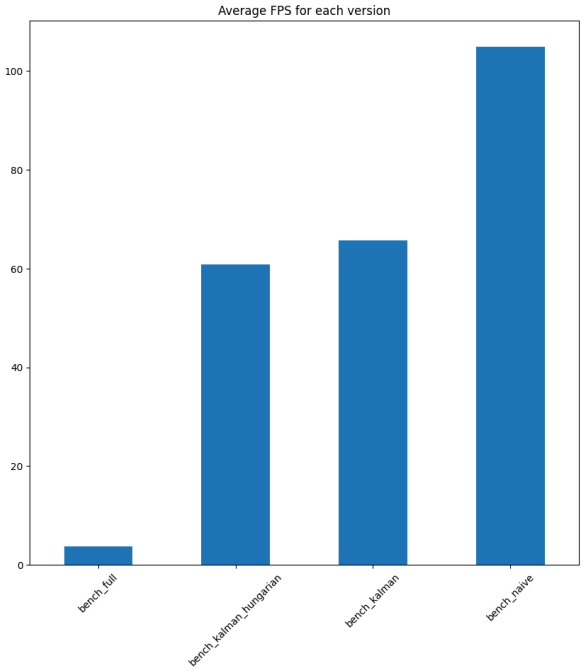
- normalisées pour ne pas avoir de différences entres les plages de valeurs de chaque métrique
- sommées et possiblement pondérées
- normalisées une fois de plus de manière à pouvoir une plage de valeurs dans $[0, 1]$ au lieu de $[0, n_{métriques}]$ afin de permettre une meilleure explicabilité des valeurs ainsi que pour faciliter un possible futur post-processing/thresholding.

3. Évaluation des résultats

Dans cette section, nous procédons à une évaluation complète des résultats de notre système de suivi d'objets, en examinant les performances à chaque étape du projet, de l'approche initiale naïve jusqu'à l'approche finale sophistiquée.

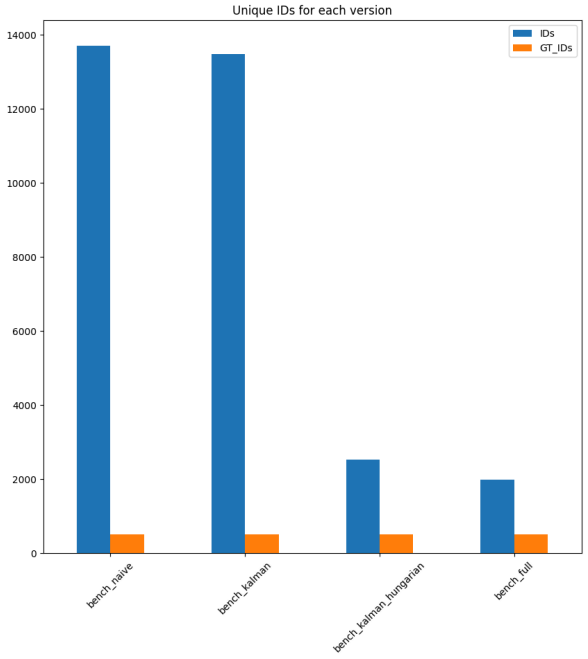
3.1 Vitesse de Suivi & Précision des tracks

- FPS Moyens: Nous mesurons la vitesse du système en termes de frames par seconde (fps) moyens, un indicateur crucial pour l'efficacité en temps réel.



Nous pouvons voir que nous sommes en capacité de faire de l’inférence en temps réel jusqu’au moment ou nous ajoutons les embeddings (avec ResNet18), ceci est dû au fait que les embeddings sont longs à calculer sur CPU, peut-être avec un GPU les résultats seraient différents.

- Nombre de Tracks Trouvés vs Réels:

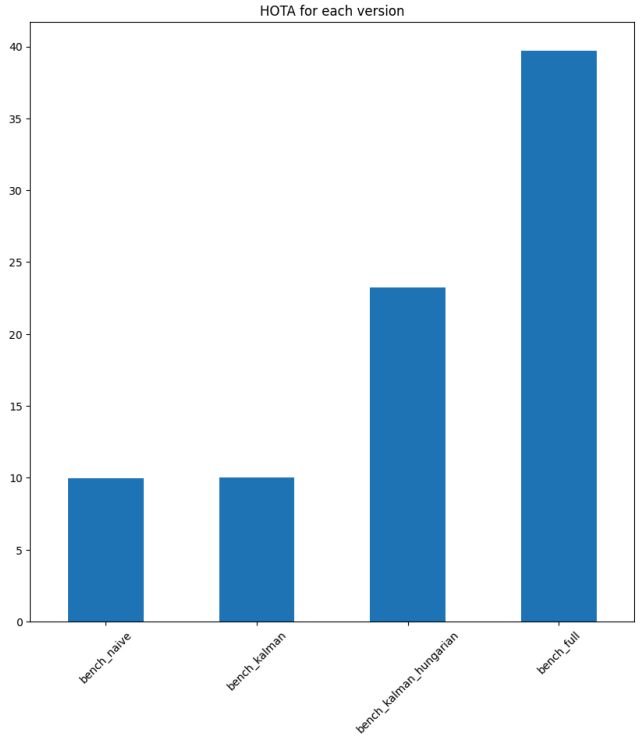


L'efficacité du suivi est également évaluée en comparant le nombre de tracks détectés par le système par rapport au nombre réel de tracks, ce qui donne une indication de la capacité du système à suivre avec précision tous les objets présents.

Ici, à l'inverse des fps moyens, la qualité augmente grandement au fur et à mesure des améliorations. On note une forte amélioration à l'ajout de l'algorithme de matching hongrois ce qui est attendu car cet élément est ajouté spécifiquement pour cet aspect.

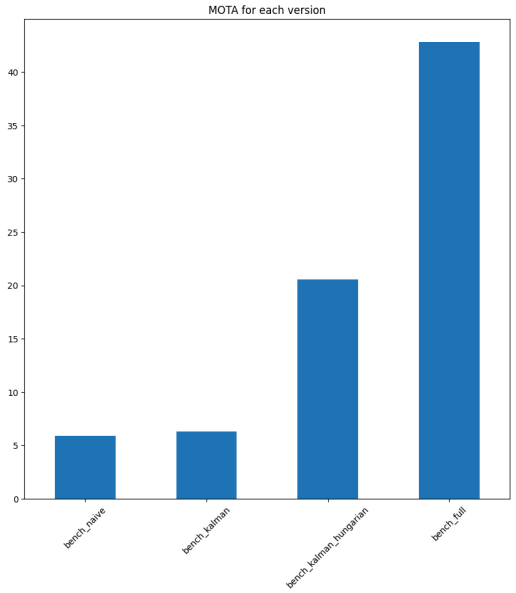
3.2. Métriques de TrackEval

- HOTA (Higher Order Tracking Accuracy): Évalue la précision globale du suivi en considérant à la fois la détection et l'association des identités.



Commençant avec un score relativement bas pour l'approche naïve (bench_naive), une légère amélioration avec l'introduction du filtre de Kalman (bench_kalman), une augmentation plus notable avec l'intégration de l'algorithme hongrois (bench_hungarian), et finalement une amélioration significative pour l'approche finale complète (bench_full).
Sachant que le meilleur score atteint sur mot-challenge-2015 en HOTA est 49.9, je suis satisfait des résultats.

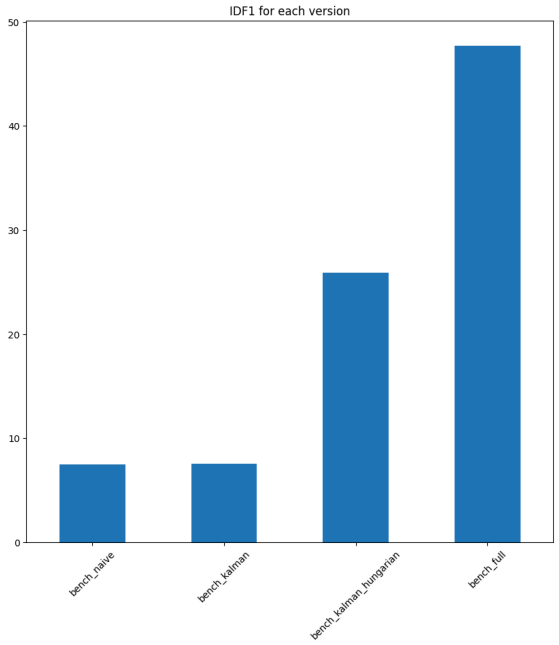
- MOTA (Multiple Object Tracking Accuracy): Mesure l'exactitude en tenant compte des erreurs de détection, des fausses alarmes et des erreurs d'association.



Le score MOTA se comporte très similairement au score HOTA avec cependant une plus grande amélioration relative sur les versions plus naïves.

Le meilleur résultat sur ce benchmark est de 62.8 ce qui correspond à 20 points de plus que mon résultat actuel. Ceci est surement dû au fait que certaines détections disparaissent pour une frame à cause de bounding boxes qui “clignotent” de part des erreurs de détection.

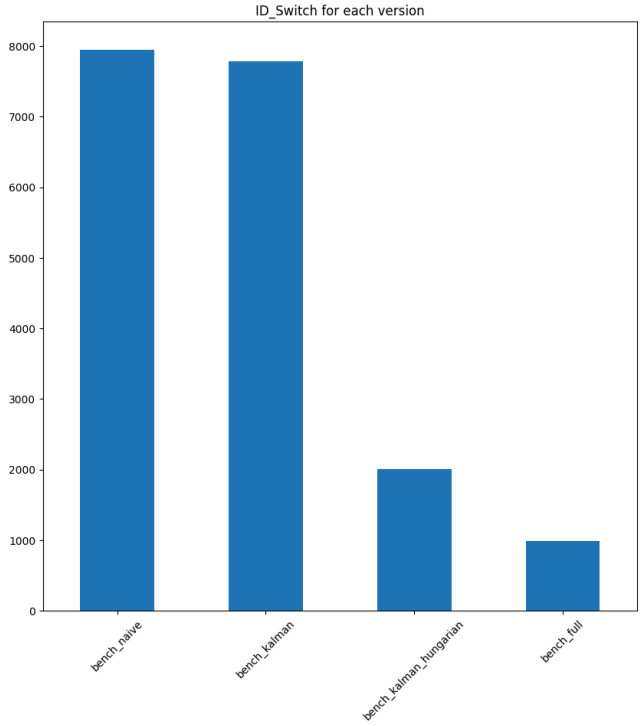
- IDF1: Évalue la fréquence à laquelle les identités sont correctement maintenues tout au long du suivi.



Nous pouvons voir que nous avons une augmentation majeure de la métrique à l’ajout de l'algorithme hongrois. Ceci est cohérent car ce dernier permet une meilleure association d’une frame à une autre. De plus, les embeddings semblent permettre une meilleure rétention des tracks grâce à une meilleure résilience aux variations de la vidéo.

Le meilleur score de ce benchmark est actuellement 64.2 ce qui n’est pas si éloigné de mes résultats.

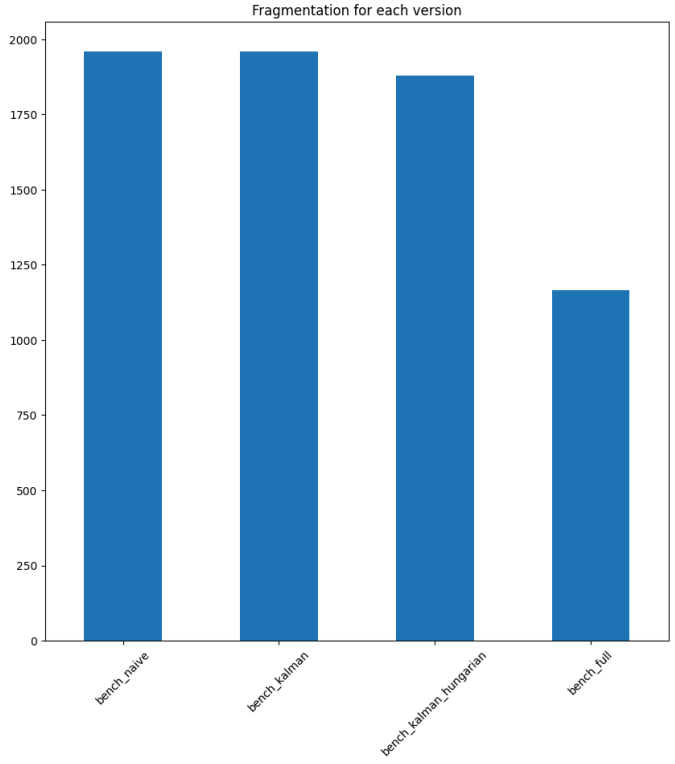
- ID_Switch: Compte le nombre de fois où une piste change incorrectement d'identité.



Les résultats des métriques précédentes est exacerbé avec cette dernière. Ici, l'ajout de l'algorithme hongrois permet de diviser par 4 le nombre de mauvais switches.

Mon Tracker a un score de 990, ce qui est toujours 600 switches en moyenne de plus que le meilleur performeur du benchmark.

- Fragmentation: mesure la fréquence à laquelle les pistes sont interrompues.



Nous touchons un point faible de mon algorithme que je n'ai pas encore vraiment amélioré, c'est l'interruption des tracks. En effet, en fonction de notre détecteur de personnes nous aurons des résultats plus ou moins robustes. Comme vous pouvez le voir, je n'ai pas réussi à le faire.

3.3 Qualité Subjective

Cette section a pour but de compléter et conclure sur l'évaluation de notre Tracker. Cette analyse subjective, basée sur l'observation des résultats de tracking, permet de juger de l'aisance avec laquelle notre système maintient l'intégrité des tracks à travers les frames, même en présence de défis tels que les occlusions ou les interactions entre les objets suivis.

Notre examen visuel des séquences de suivi révèle que l'ajout de l'algorithme hongrois et des embeddings ResNet18 a sensiblement augmenté la cohérence et la stabilité des identités des objets au fil du temps. Les objets sont suivis avec une plus grande précision, et les erreurs d'identité sont moins fréquentes, résultant en un suivi visuellement plus fluide et fiable.

Bien que la fragmentation reste un défi, il est clair que les améliorations successives ont rendu notre système de suivi plus robuste face aux variations dynamiques des scènes. En définitive, notre système de suivi, bien que perfectible, démontre une performance satisfaisante qui confirme le potentiel des méthodes que nous avons intégrées pour le suivi d'objets en temps réel.

Un point reste à traiter: l'interruption et la reprise de tracks qui se fait beaucoup trop fréquemment.
Ce problème apparaît dans deux situations principalement:

- les zones avec de nombreuses personnes serrées (tâche difficile de base)
- les détections de ce tracks qui disparaissent pour une seule frame (mais qui fait que notre track s'arrête et un nouveau est initialisé la frame suivante)

4. Perspectives d'améliorations

Dans le but d'affiner davantage notre système de suivi et d'atteindre une performance optimale, plusieurs axes d'amélioration sont envisagés pour l'avenir :

4.1 Association de Tracks sur Plusieurs Frames

Une piste d'amélioration consisterait à étendre l'association des tracks au-delà de la dernière frame analysée. En considérant un historique plus long, potentiellement les 10 dernières frames, notre système pourrait mieux comprendre la trajectoire des objets et réduire les erreurs d'identité, en particulier dans les cas d'occlusions temporaires ou de mouvements rapides.

4.2 Exploitation des Ressources GPU

L'utilisation d'un GPU pourrait considérablement accélérer le calcul des embeddings, qui constitue actuellement un goulot d'étranglement en termes de vitesse d'inférence. Ceci permettrait de maintenir la performance en temps réel tout en exploitant la richesse des informations fournies par les embeddings ResNet18. Alternativement, l'expérimentation avec des modèles plus légers tels que MobileNet pourrait également s'avérer bénéfique, en offrant un bon compromis entre vitesse et précision.

4.3 Refonte du Code pour Plus de Modularité

Le refactoring du code pour augmenter sa modularité permettrait une meilleure évolutivité du système et faciliterait l'intégration de nouvelles améliorations. Un code bien structuré et modulaire est essentiel pour permettre une adaptation rapide aux avancées futures du domaine et pour expérimenter avec de nouvelles techniques ou architectures plus performantes.

5. Conclusion

Ce rapport a présenté l'élaboration et l'évaluation détaillée d'un système de tracking de personnes.
Nous avons mis en lumière les améliorations significatives apportées par l'introduction de filtres de Kalman, l'algorithme hongrois, et l'usage des embeddings ResNet18, ainsi que par l'application de techniques de détection avancées telles que YOLOv7. La progression mesurable à travers les métriques de suivi telles que HOTA, MOTA, IDF1, ainsi que l'analyse qualitative du suivi, confirment l'efficacité des méthodes employées.

L'évaluation a révélé que, si l'inférence en temps réel est atteignable, des défis subsistent, notamment en ce qui concerne le calcul des embeddings sur CPU et la fragmentation des pistes. Cela ouvre la voie à des pistes d'amélioration concrètes, incluant l'optimisation du traitement par GPU et une extension de l'association de tracks sur plusieurs frames pour une meilleure résilience et précision.

En conclusion, le travail accompli jusqu'à présent fournit une base solide pour le suivi d'objets et établit un cadre pour les améliorations futures.

Appendix

Diagramme de notre algorithme