Intercorrélation basée apprentissage profond pour le suivi multi-cibles

Pierre Marigo[1,2] Jérôme Thomas[1] Claire Labit-Bonis[1] Frédéric Lerasle[2,3]

[1] ACTIA Automotive, 5 rue Jorge Semprun, 31432 Toulouse
[2] CNRS, LAAS, 7 avenue du Colonel Roche, 31400 Toulouse
[3] Université de Toulouse, UPS, 118 route de Narbonne, 31400 Toulouse

{pierre.marigo, jerome.thomas, claire.labit-bonis}@actia.fr, lerasle@laas.fr

Résumé

Suivre plusieurs cibles en mouvement dans un flux vidéo implique de savoir localiser et réidentifier ces cibles dans des scènes variées et parfois encombrées. Nous présentons une méthode de suivi par détection avec recalage visuel multi-hypothèses basé sur une intercorrélation par apprentissage profond. En plus d'un unique réseau de neurones tout-en-un servant à la détection et au suivi des cibles, nous mettons en avant une stratégie de gestion des trajectoires par caractérisation de leurs états. Nous démontrons des performances à l'état de l'art comparativement à la littérature, au travers d'une analyse quantitative et qualitative sur le challenge de référence MOT17.

Mots Clef

Suivi par détection multi-cibles, intercorrélation, apprentissage profond

Abstract

Tracking multiple moving targets in a video stream requires localizing and re-identifying them in varied and sometimes cluttered scenes. We present a tracking-by-detection method with multi-hypothesis visual pose estimation based on deep learning cross-correlation. In addition to a single all-in-one neural network for target detection and tracking, we also put forward a strategy for managing trajectories by characterizing their states. We demonstrate the quality of our approach and the gains made by comparing ourselves to the literature on the MOT17 benchmark through a quantitative and qualitative analysis.

Keywords

Multiple object tracking-by-detection, cross-correlation, deep learning

1 Introduction

Suivi par détection multi-cibles. Le suivi multi-cibles (*Multiple Object Tracking, MOT*) [18] est couramment utilisé dans de nombreuses applications, notamment dans les domaines de la vidéo-surveillance ou plus récemment l'assistance à la conduite. Ces approches ont pour objectif l'identification et le maintien des trajectoires de différentes cibles, dans des flux vidéos.

Le suivi par détection (*tracking-by-detection*) est une stratégie de suivi qui fait largement consensus dans la communauté Vision par Ordinateur. Lorsqu'il est appliqué en ligne *i.e.* pour une inférence à l'instant courant et avec seule connaissance des instants passés, il nécessite, (i) de détecter les cibles présentes dans l'image courante, puis (ii) d'associer ces détections aux trajectoires des cibles identifiées dans les images précédentes. Ces deux parties sont complémentaires : le détecteur guide le traqueur vers les détections pour retrouver les cibles dans la nouvelle image et limite les dérives, quand le traqueur permet de compenser d'éventuelles erreurs du détecteur.

Le MOT Challenge [18] est un *challenge* de référence dans ce domaine. Il permet de mesurer la qualité d'un traqueur de piétons dans un ensemble de scènes complexes et variées mises à disposition, et ainsi de comparer les approches de la communauté MOT sur une base commune.

Approches existantes. Les traqueurs intègrent une étape de prédiction du déplacement inter-image de chacune des cibles. Cette étape utilise régulièrement des solutions de type filtrage de Kalman [10] ou Constant Velocity Model qui estiment la nouvelle position de la cible en se basant sur l'historique de ses positions dans l'image. Si ces solutions peuvent être pertinentes pour des situations simples, elles peuvent poser problème dans des cas plus complexes comme lors d'un changement brutal de direction. En début d'algorithme, elles prédisent donc « en aveugle » les positions des cibles suivies sans s'appuyer sur le contenu de l'image courante. Le suivi visuel mono-cible (*Visual Object Tracking, VOT* [11]) consiste à suivre une unique cible d'intérêt dans une vidéo en connaissant seulement sa position initiale à l'instant t_0. Depuis GOTURN [9], la communauté VOT a proposé plusieurs traqueurs basés sur de l'apprentissage profond utilisant des réseaux siamois pour caractériser la signature des cibles et les réidentifier d'une image à l'autre. La pertinence de l'utilisation de caracté-

ristiques profondes pour les tâches de réidentification et de suivi visuel a été démontrée par de nombreux travaux ([11], [12]), si bien que la grande majorité des nouvelles approches utilise désormais ces méthodes par apprentissage profond. Bien qu'aujourd'hui certains couples détecteurs-traqueurs de l'état de l'art comme ByteTrack [30] n'exploitent pas ces avancées récentes, plusieurs approches MOT intègrent un recalage visuel par apprentissage profond (OSRR [13], LSST [7]). Ces approches montrent que, à détecteur équivalent, recaler les positions en considérant d'emblée les informations visuelles offertes par chaque image apporte un gain substantiel dans les performances. Cependant, ces traqueurs nécessitent l'apprentissage d'un second réseau de neurones pour le recalage et/ou la réidentification des cibles, s'ajoutant donc à celui du détecteur.

L'approche récente FairMOT [31] pallie la nécessité d'avoir deux réseaux distincts pour la détection et la réidentification en proposant un réseau tout-en-un. A chaque instant image, c'est-à-dire pour chaque image issue du flux vidéo, son architecture infère la position des cibles, mais extrait également leur signature visuelle individuelle grâce à un réseau entraîné à la fois à localiser des cibles (tâche dite de détection), et à en distinguer des instances (tâche dite de réidentification). FairMOT utilise cependant un recalage inter-image classique *via* un filtre de Kalman pour inférer la position des cibles en début d'algorithme, avec les limites présentées précédemment.

En s'inspirant du réseau de détection de FairMOT ainsi que des avancées observées en VOT, nous proposons un nouveau traqueur nommé C2DT (*Combined Cross-correlation-based Detector and Tracker*) permettant d'associer un réseau tout-en-un à des techniques de recalage visuel et d'intercorrélation basées apprentissage profond (SiameseRPN [14], SiamFC++ [27]), sans toutefois nécessiter de nouvel apprentissage.

Nos contributions. Eu égard à ces constats, nous mettons en avant plusieurs contributions dans le contexte du suivi visuel multi-cibles :

- Le développement d'une approche originale multi-hypothèses utilisant l'intercorrélation par apprentissage profond pour réidentifier les cibles entre images successives ;
- L'intégration de cette approche de suivi par recalage visuel dans un couple détecteur-traqueur basé sur un unique réseau de neurones ;
- La mise en place d'une stratégie de gestion des trajectoires par leurs états dans le flux ;
- L'évaluation de notre approche sur le *challenge* de référence MOT17, démontrant des performances de suivi des cibles au niveau de l'état de l'art.

Dans la suite, la section 2 mentionne les principales méthodes utilisées pour le suivi de cibles. Nous décrivons ensuite le fonctionnement de notre traqueur en section 3. Enfin, nous évaluons notre approche et montrons les gains apportés par chacune de ses composantes au travers d'une étude par ablations dans la section 4.

2 État de l'art

La communauté Vision est très active sur la problématique du suivi visuel multi-cibles en ligne. La variabilité des vidéos (caméras statiques *vs.* mobiles, environnement contrôlé ou non, encombrement, etc.) représente un véritable défi : les traqueurs doivent rester performants dans des scènes variées, tout en gérant des occultations, des erreurs de détection, ou encore des cibles de même apparence. Nous présentons ici les principales approches répondant à cette problématique.

2.1 Suivi par détection multi-cibles en ligne

Approches en ligne *vs.* hors ligne. Le suivi des cibles peut être fait en ligne, *i.e.*, au fil du flux en n'utilisant que les informations obtenues jusqu'à l'instant courant, ou bien hors ligne, *i.e.* en utilisant une fenêtre temporelle large d'images, voire la séquence entière. De nombreuses applications nécessitent de traiter les flux vidéo en ligne ; aussi privilégions-nous ce contexte et comparons notre approche aux soumissions en ligne du *challenge* MOT17.

Reconstruction de trajectoires. L'algorithme 1 résume le principe général du suivi multi-cibles par détection. Image par image, l'objectif est de reconstruire les trajectoires de plusieurs cibles en mouvement. Plus précisément, il s'agit (i) de détecter les cibles dans chaque image, c'est-à-dire leurs coordonnées x, y, w, h (resp. centre, largeur et hauteur) en pixels image, (ii) de prédire la position courante des trajectoires déjà existantes, puis (iii) d'associer ces trajectoires avec les détections issues du détecteur. Pour chaque détection non associée, une nouvelle trajectoire est créée. Chaque traqueur a généralement une stratégie de gestion des trajectoires qui met à jour leur état (position, apparence, destruction, etc.) en fin de processus.

Algorithme 1 Suivi par détection en ligne multi-cibles

Require: $\{\mathcal{I}_{1:N}\}$ un ensemble de N images consécutives
Ensure: $\{\mathcal{T}_{1:N}^{1:M}\}$ un ensemble de M trajectoires à chaque instant image
$\{\mathcal{T}_0\} \leftarrow \emptyset$
for $t \in [1, N]$ **do**
  $\{\mathcal{D}_t\} \leftarrow$ **détecter**$(\{\mathcal{I}_t\})$ ▷ Détection des cibles dans $\{\mathcal{I}_t\}$
  if $\mathcal{T}_{t-1} \neq \emptyset$ **then**
    $\{\mathcal{H}_t\} \leftarrow$ **prédire**$(\{\mathcal{T}_{t-1}\})$ ▷ Inférence des nouvelles positions
    $\{\mathcal{T}_t\}, \{\mathcal{D}_{t_{rem}}\} \leftarrow$ **associer**$(\{\mathcal{H}_t\}, \{\mathcal{D}_t\})$ ▷ Association
  if $\{\mathcal{D}_{t_{rem}}\} \neq \emptyset$ **then**
    $\{\mathcal{T}_t\} \leftarrow \{\mathcal{T}_t\} \cup \{\mathcal{D}_{t_{rem}}\}$ ▷ Création des trajectoires
  $\{\mathcal{T}_t\} \leftarrow$ **mettre_à_jour**$(\{\mathcal{T}_t\})$ ▷ Maintien, destruction, *etc.*

2.2 Détection des cibles

La première étape dans l'algorithme 1 de suivi par détection concerne la détection en ligne des cibles. Bien que le détecteur (**détecter**$(\{\mathcal{I}_t\})$) ne soit pas l'objet principal de cet article, il est nécessaire de souligner son importance déterminante dans les résultats globaux au MOT Challenge : plusieurs articles comme Tracktor [1] tentent d'ailleurs de faire passer le message *"A detector is all you need"*.

Dans le MOT Challenge, deux catégories sont présentes :

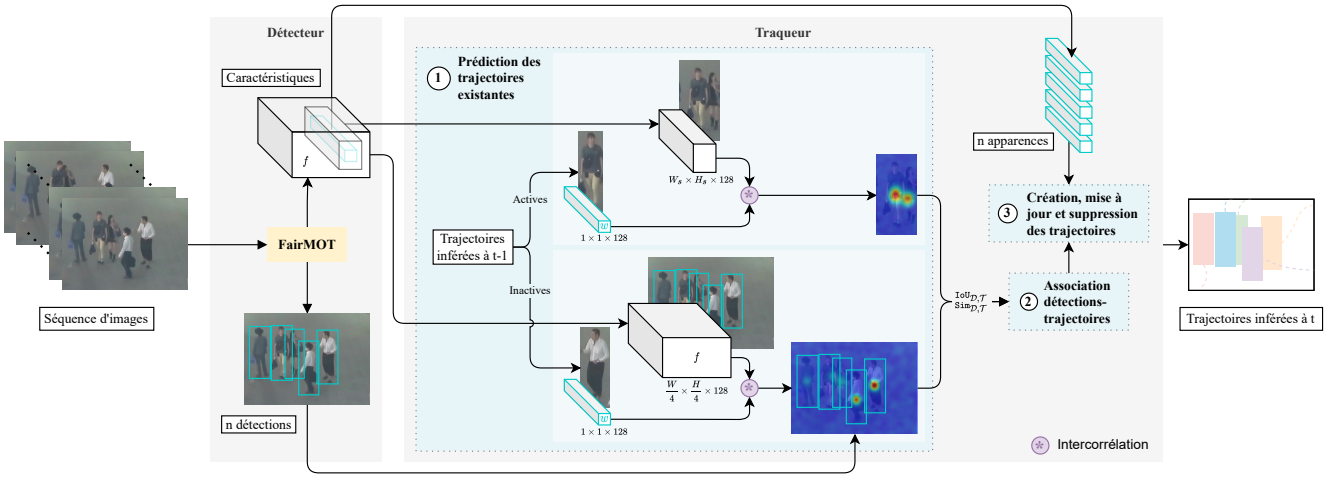

FIGURE 1 – Processus complet du suivi par détection multi-cibles. Le détecteur entraîné par FairMOT [31] est appliqué sur chacune des images en entrée. Ce réseau renvoie un ensemble de détections et leurs vecteurs d'apparence associés issus de la carte de caractéristiques de l'image f. Après une étape de prédiction de la position des trajectoires existantes via l'intercorrélation ①, les détections courantes sont associées à ces trajectoires ②. Enfin, l'apparence et les attributs des trajectoires sont mis à jour en fin de processus ③.

l'une utilise des détections publiques et l'autre des détections privées. Souhaitant baser notre approche sur l'utilisation du réseau détecteur tout-en-un développé par FairMOT, nous nous plaçons dans la catégorie des détections privées. Néanmoins, comparer uniquement la composante traqueur de cette catégorie n'est pas tâche aisée, tant l'impact du détecteur est important. Dans cet article, nous cherchons à montrer que, à détecteur identique, notre traqueur permet d'améliorer le suivi de cibles dans un flux vidéo.

2.3 Prédiction des positions des cibles

De nombreuses approches ont abordé le problème de l'étape de prédiction de la position des cibles suivies (**prédire**($\{\mathcal{T}_{t-1}\}$)). Ainsi, Tracktor [1] utilise un Constant Velocity Model, tandis que EAMTT [20] prédit les positions des cibles à suivre à l'aide d'un filtre à particules, et que de nombreux traqueurs utilisent un filtre de Kalman [10] pour cette même tâche ([26], [30], [31]). Certaines de ces approches ([21], [17]) peuvent générer plusieurs hypothèses de position pour chacune des cibles, mais considèrent uniquement les positions passées des cibles et non pas leur apparence. En contexte mono-cible (VOT), de nombreuses approches utilisent des réseaux siamois profonds, notamment depuis GOTURN [9], pour obtenir un recalage visuel plus précis. Certaines méthodes comme SiamFC++ [27] ou SiameseRPN [14] utilisent par ailleurs l'opération d'intercorrélation pour estimer les déplacements des cibles à partir des sorties des réseaux profonds. Récemment intégrés en contexte multi-cibles avec LSST [7] ou OSRR [13], les traqueurs visuels siamois en contexte MOT permettent un recalage des trajectoires plus performant. Ces approches nécessitent cependant d'apprendre un réseau de neurones dédié aux tâches de suivi en plus de celui dédié à la détections des cibles.

2.4 Association trajectoires/détections

A chaque instant image, la phase d'association (**associer**($\{\mathcal{H}_t\}, \{\mathcal{D}_t\}$)) prend généralement en compte plusieurs éléments qui permettent de définir une matrice de coûts pour chaque association potentielle. En plus de la position précédente des cibles, de nombreuses approches utilisent la similarité d'apparence entre les détections et les trajectoires pour qualifier la ressemblance des associations testées. Ce critère de similarité repose généralement sur des représentations propres aux cibles et issues d'un réseau de neurones entraîné pour cette tâche ([1], [26]). OSRR [13] utilise par exemple un seul réseau siamois, entraîné à la fois pour l'estimation de la nouvelle position image et pour la génération des signatures individuelles. FairMOT [31] intègre une branche dédiée à la réidentification à son réseau de détection ; chaque cible détectée est associée à son vecteur descripteur d'apparence.

2.5 Stratégie de gestion des trajectoires

Habituellement, chaque trajectoire possède des attributs qui lui sont associés et mis à jour au cours de la vidéo : position, modèle d'apparence, gestion de sa naissance et de sa destruction, statut d'inactivité, etc. (**mettre_à_jour**($\{\mathcal{T}_t\}$)). Certains traqueurs mettent à jour la position et le modèle d'apparence de manière progressive ([7], [13], [31], [1]). Régulièrement, les trajectoires ont un statut dit actif ou inactif, selon si elles ont été associées à une détection lors de la dernière phase ou non. Beaucoup de traqueurs ([26], [1], [13], etc.) considèrent les trajectoires inactives depuis trop longtemps comme perdues et les détruisent. Afin de compenser d'éventuelles erreurs du détecteur, certains traqueurs comme OSRR [13] utilisent le recalage visuel pour maintenir des trajectoires même en l'absence de détection associée.

3 Notre approche

La section précédente nous amène aux constats et aux perspectives suivants :

1. en contexte MOT, au contraire des traqueurs visuels basés sur des réseaux profonds, l'utilisation d'un réseau de neurones tout-en-un nous permet d'intégrer un recalage visuel sans apprentissage supplémentaire ;
2. les approches de suivi mono-cible (*Single Object Tracking, SOT*) impliquant un recalage visuel basé sur l'intercorrélation montrent des performances accrues, que nous intégrons dans une approche MOT. Grâce à l'utilisation du vecteur d'apparence produit par le réseau tout-en-un de FairMOT, nous proposons une approche multi-hypothèses basée sur cette opération d'intercorrélation pour inférer la position des cibles suivies à l'étape de prédiction ;
3. en plus de notre propre stratégie de gestion des trajectoires, l'intercorrélation basée apprentissage profond offre une sortie riche que nous exploitons *via* une nouvelle méthode pour maintenir les trajectoires en activité.

La section suivante détaille le fonctionnement de notre approche ainsi que l'intégration des éléments précédemment cités *via* une représentation et une stratégie originales de gestion des trajectoires.

3.1 Représentation des trajectoires

Position image. A tout instant, notre traqueur a connaissance de la position image de chacune des trajectoires existantes, *i.e.* leurs coordonnées x, y, w, h.

Modèle d'apparence. Le réseau de détection de FairMOT prédit en parallèle les coordonnées des cibles détectées, ainsi qu'un vecteur de 128 caractéristiques propre à chacune d'elle. Nous définissons alors le modèle d'apparence d'une trajectoire à partir de ce vecteur et le mettons à jour au fur et à mesure. Bien que de dimensions $1 \times 1 \times 128$, le vecteur d'apparence fourni par le réseau FairMOT agrège l'ensemble des caractéristiques ayant permis de réidentifier la cible et donc une information visuelle discriminante.

Statut des trajectoires. Chaque trajectoire peut être soit active, soit inactive. Ce statut, mis à jour à chaque instant image, indique si la trajectoire a été associée à une détection ou non. Une trajectoire nouvellement créée est considérée comme non confirmée tant que plusieurs détections successives ne lui sont pas associées. Si aucune détection n'est associée à une trajectoire non confirmée, elle est supprimée.

Score de confiance. Nous attribuons un score de confiance à toutes les trajectoires, quel que soit leur statut. Compris entre 0 et 1, ce score est mis à jour après chaque association. Ce score de confiance nous permet de maintenir la trajectoire active comme décrit en section 3.4.

3.2 Intercorrélation 2D normalisée basée apprentissage profond

A la différence des approches qui prédisent le déplacement des trajectoires en se basant uniquement sur leurs positions ([1], [26], [30], [31]), nous retrouvons les cibles suivies dans chaque nouvelle image grâce à un recalage visuel. Contrairement à OSRR et LSST qui utilisent un deuxième réseau pour le faire, nous utilisons directement les informations d'apparence issues du réseau FairMOT.

Réidentification dans FairMOT. Au-delà des détections, FairMOT inclut une tête supplémentaire de réidentification qui produit une carte de caractéristiques de l'ensemble de l'image traitée (*cf.* [31] pour plus de détails sur l'architecture du réseau). Ces caractéristiques sont optimisées lors de l'apprentissage pour réidentifier chacune des cibles. De ce fait, pour une image de dimensions $W \times H \times 3$, nous obtenons une carte de caractéristiques f de dimensions $\frac{W}{4} \times \frac{H}{4} \times 128$ entraînée spécifiquement pour la similarité d'apparence.

Zone de recherche. Afin de propager les trajectoires des cibles suivies, nous devons retrouver leur signature visuelle dans les nouvelles images issues du flux vidéo. Le déplacement des cibles inter-images étant limité, nous définissons, à l'instar de nombreux traqueurs, une zone de recherche centrée sur chaque trajectoire active et fonction de sa taille. Ainsi, la zone de recherche considérée est de dimensions $W_s \times H_s$.

Carte de similarité. Connaissant la zone de recherche, nous calculons alors un score de similarité entre la signature visuelle w de la cible recherchée et les pixels de f à l'intérieur de cette zone. A l'inverse des traqueurs prédisant une unique position ([13], [9]), ou d'approches mesurant un seul score d'association pour chaque couple détection/trajectoire ([31]), nous obtenons en sortie $W_s \times H_s$ scores dans le voisinage du centre de notre trajectoire. Cette carte de similarité nous permet ainsi d'adopter une stratégie multi-hypothèses (*cf.* section 3.3).

Métrique de similarité. La similarité cosinus est une mesure régulièrement utilisée en suivi multi-cibles pour mesurer la similarité entre deux vecteurs d'apparence ([13], [31], [26]). Pour prendre en compte les différences des dynamiques dans les caractéristiques, nous faisons le choix de centrer et réduire nos vecteurs. Au final, pour toute position (x, y) dans la zone de recherche, l'opération mesurant la similarité entre les caractéristiques f de l'image et le vecteur d'apparence w de la cible s'écrit :

$$S(x,y) = \frac{\sum_{k=0}^{127} (f(x,y,k) - \bar{f}(x,y)) \times (w(k) - \bar{w})}{\sigma_f \times \sigma_w} \quad (1)$$

avec $f(x, y, k)$ la valeur du canal k de f à la position (x, y), $w(k)$ la valeur du canal k du vecteur d'apparence w, σ_f et σ_w les écarts-types respectivement des caractéristiques f de l'image et de la signature visuelle w. $\bar{w}$ dénote

la moyenne des valeurs de la signature visuelle et $\bar{f}(x, y)$ celle des 128 caractéristiques de l'image au pixel étudié.

Intercorrélation. L'utilisation de cette similarité cosinus revient à calculer l'intercorrélation normalisée entre le vecteur d'apparence et la zone de recherche. Contrairement à l'opération classique d'intercorrélation 2D utilisée pour retrouver une sous-image dans une image globale, nous considérons uniquement ce vecteur d'apparence pour l'intercorrélation car il inclut toutes les caractéristiques permettant de réidentifier la cible. Si l'intercorrélation a déjà été utilisée par le passé ([28]), elle s'appuie ici sur des descripteurs d'apparence dans $\mathbb{R}^{128}$ optimisés lors de l'apprentissage, et non pas sur les pixels bruts de l'image, permettant ainsi un recalage visuel de la cible plus précis. Cette intercorrélation basée apprentissage profond normalisée nous fournit donc, pour chaque position spatiale de la zone de recherche, un critère de similarité, *i.e.* un niveau de confiance de la présence de la cible pour chaque position.

3.3 Prédiction et association

La figure 1 illustre le déroulement global de notre méthode. A chaque instant, le traqueur doit associer les détections et les informations provenant de l'étape de recalage visuel pour reconstruire les trajectoires des cibles. Le statut d'activité des trajectoires traduit leur niveau de certitude : une trajectoire active est confiante ; une trajectoire inactive peut quant à elle dériver, *e.g.* en raison d'occultations ou d'erreurs de détection. Certaines méthodes comme DeepSORT ou Tracktor priorisent alors l'association des trajectoires actives en les traitant avant celles inactives. Nous proposons une stratégie d'association globale des trajectoires aux détections à chaque instant image, *i.e.* des trajectoires actives et inactives confondues. Ainsi, nous ne favorisons pas l'association des trajectoires actives sur celle des inactives, contrairement aux approches précédemment citées.

Trajectoires actives et stratégie multi-hypothèses. Durant l'étape de prédiction (*cf.* figure 1), l'intercorrélation 2D normalisée centrée (*cf.* section 3.2) appliquée aux trajectoires actives produit une carte de similarité. Comme illustré sur la figure 2, pour chaque trajectoire $\mathcal{T}$ active, nous obtenons donc un ensemble de $W_s \times H_s$ scores avec $W_s \times H_s$ la taille de la zone de recherche autour de la trajectoire considérée (*cf.* section 3.2). Sur cette carte de similarité, nous localisons les points à forte similarité (potentiellement plusieurs) grâce à un algorithme de recherche de maxima locaux. Nous envisageons alors plusieurs hypothèses de déplacement de la trajectoire $\mathcal{T}$, en juxtaposant artificiellement une boîte $\mathcal{T}_i$ de même taille que la boîte englobante de $\mathcal{T}$ et centrée sur chaque maximum ainsi obtenu. Un score $\texttt{IoU}_{\mathcal{D},\mathcal{T}_i}$ est ensuite calculé entre chaque boîte posée $\mathcal{T}_i$ et chaque détection $\mathcal{D}$. Pour chaque couple trajectoire/détection, nous sélectionnons la meilleure hypothèse en définissant $\texttt{IoU}_{\mathcal{D},\mathcal{T}} = \max(\texttt{IoU}_{\mathcal{D},\mathcal{T}_i})$.
Connaissant la boîte $\mathcal{T}_i$ conservée pour $\texttt{IoU}_{\mathcal{D},\mathcal{T}}$, nous définissons $\texttt{Sim}_{\mathcal{D},\mathcal{T}}$ comme le maximum local de la carte de similarité correspondant à $\mathcal{T}_i$ issue de l'intercorrélation. Celle-ci étant normalisée, $\texttt{Sim}_{\mathcal{D},\mathcal{T}} \in [0, 1]$ traduit alors la similarité entre l'hypothèse retenue et la cible suivie. Dit autrement, nous autorisons un déplacement dans des directions pouvant être potentiellement opposées (*cf.* figure 2). Cette stratégie multi-hypothèses est donc maintenue jusqu'à l'application de l'algorithme d'association trajectoires/détections, qui prendra la décision finale en considérant un coût unique pour chaque association trajectoire/détection possible.

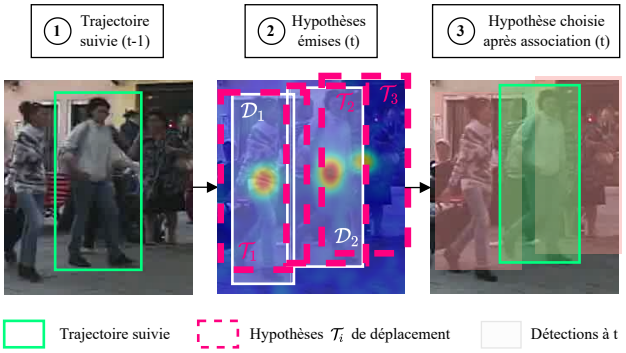

FIGURE 2 – Stratégie multi-hypothèses lors de la prédiction des positions des trajectoires actives. Pour chaque trajectoire active ①, la carte de similarité issue de l'intercorrélation permet d'émettre plusieurs hypothèses $\mathcal{T}_i$ centrées sur les maxima locaux ②. L'étape d'association choisira la meilleure hypothèse pour l'ensemble des trajectoires, au regard des détections $\mathcal{D}_j$ disponibles ③.

Trajectoires inactives. Une trajectoire inactive peut être perdue depuis plusieurs instants image, *i.e.* une détection sur cette cible à $t + n$ pourrait être éloignée de sa dernière position connue. Les approches SOT reposant sur l'intercorrélation étendent la zone de recherche autour de la cible lorsqu'elles l'ont perdue, mais cette stratégie serait particulièrement coûteuse en temps de calcul pour notre traqueur en contexte MOT au vu du grand nombre de cibles à suivre. Notre but étant toujours d'associer une trajectoire à une détection, nous considérons que le déplacement qu'a pu effectuer une trajectoire inactive à chaque image dépend de sa largeur $L_{\mathcal{T}}$. Nous autorisons donc une trajectoire inactive à s'associer avec toutes les détections à une distance de d pixels telle que :

$$d \leq n * \xi * L_{\mathcal{T}} \tag{2}$$

avec n la durée d'inactivité de la trajectoire. Nous intercorrélons ensuite le pixel d'apparence de la trajectoire inactive avec les zones de recherche autour des détections candidates (*cf.* figure 1). Nous posons alors une boîte de la taille de la cible sur le maximum de chaque carte de similarité obtenue, et définissons un score de superposition $\texttt{IoU}_{\mathcal{D},\mathcal{T}}$ pour chaque association possible entre une trajectoire $\mathcal{T}$ inactive et une détection $\mathcal{D}$, de même qu'un score de similarité $\texttt{Sim}_{\mathcal{D},\mathcal{T}}$ égal à la valeur de ce maximum. De la même manière que pour les trajectoires actives, les scores $\texttt{IoU}_{\mathcal{D},\mathcal{T}}$ et $\texttt{Sim}_{\mathcal{D},\mathcal{T}}$ produits à partir de l'intercorrélation traduisent la cohérence spatiale et visuelle des associations.

Association trajectoires/détections. Le traitement différencié des trajectoires actives et inactives permettant d'obtenir des scores de superposition et similarité aux dynamiques identiques, nous traitons toutes les trajectoires ensemble en phase d'association. Plus précisément, nous considérons le niveau de superposition $\texttt{IoU}_{\mathcal{D},\mathcal{T}}$, donnant une information sur la distance entre une trajectoire $\mathcal{T}$ et une détection $\mathcal{D}$ et sur leur taille, ainsi que la similarité $\texttt{Sim}_{\mathcal{D},\mathcal{T}}$ entre la cible suivie et celle détectée. Finalement, le coût d'association $\mathcal{C}_{\mathcal{D},\mathcal{T}}$ d'une trajectoire avec une détection est défini par l'équation (3) :

$$\begin{aligned} \mathcal{S}_{\mathcal{D},\mathcal{T}} &= \lambda\ \texttt{Sim}_{\mathcal{D},\mathcal{T}} + (1-\lambda)\ \texttt{IoU}_{\mathcal{D},\mathcal{T}} \\ \mathcal{C}_{\mathcal{D},\mathcal{T}} &= 1 - \mathcal{S}_{\mathcal{D},\mathcal{T}} \end{aligned} \tag{3}$$

Nous appliquons ensuite un algorithme glouton sur la matrice de coûts obtenue afin d'inférer les associations détections-trajectoires à l'instant courant. A l'instar de FairMOT [31], les trajectoires et détections restantes sont associées en considérant leurs scores $\texttt{IoU}_{\mathcal{D},\mathcal{T}}$; une dernière étape d'association également basée sur $\texttt{IoU}_{\mathcal{D},\mathcal{T}}$, est réalisée avec les trajectoires non confirmées.

3.4 Stratégie de gestion des trajectoires

Score de confiance des trajectoires. Nous construisons un score de confiance de trajectoire inspiré de LSST et OSRR et adapté à notre approche, qui nous permet d'évaluer la certitude du traqueur. Ainsi, lors de l'étape d'association, nous mettons à jour son score de confiance $\mathcal{T}_{c_t}$ en fonction de sa confiance précédente $\mathcal{T}_{c_{t-1}}$, des résultats de l'intercorrélation et du score du détecteur $\mathcal{D}_c$. Pour une trajectoire non associée, son score de confiance décroît selon un facteur multiplicateur constant. Au final, la confiance d'une trajectoire est mise à jour selon l'équation (4) :

$$\mathcal{T}_{c_t} = \begin{cases} \frac{\mathcal{T}_{c_{t-1}} + \alpha\ \texttt{IoU}_{\mathcal{D},\mathcal{T}} + \beta\ \texttt{Sim}_{\mathcal{D},\mathcal{T}} + \gamma\ \mathcal{D}_c}{2} & \text{si associée} \\ \mathcal{T}_{c_{t-1}} \times \rho, & \text{sinon} \end{cases} \tag{4}$$

avec $\alpha+\beta+\gamma = 1$. Notons que certains traqueurs incluent ce score de confiance dans leur coût d'association. Il a été observé empiriquement que, pour notre traqueur, les résultats étaient meilleurs lorsque ce score de confiance n'était pas considéré au moment de l'association.

Mise à jour du statut des trajectoires. Une trajectoire associée à une détection est considérée active. A l'inverse, sans détection à lui associer, cette trajectoire devient inactive sauf si le traqueur est en mesure de maintenir son statut (*cf.* explication détaillée au paragraphe "Maintien des trajectoires actives non associées"). Une trajectoire inactive à laquelle on associe une détection redevient active. Le statut d'(in)activité des cibles suivies permet de gérer des cas de non-détection ou d'occultation, en préférant qualifier la trajectoire d'inactive, sans pour autant la détruire. Toutefois, si une trajectoire reste inactive pendant plus de n images consécutives, elle est définitivement détruite.

Mise à jour de l'apparence des trajectoires. Chaque trajectoire est représentée par un unique vecteur d'apparence $\mathcal{T}_A$, de dimensions $1 \times 1 \times 128$, qui correspond à l'agrégation au cours du temps des caractéristiques de réidentification $\mathcal{D}_A$ issues de la branche de réidentification du réseau FairMOT. En d'autres termes, nous mettons à jour ce vecteur d'apparence $\mathcal{T}_A$ de manière progressive quand une association est faite, en utilisant la signature visuelle de la détection $\mathcal{D}_A$ et selon l'équation (5) :

$$\mathcal{T}_A = \omega\ \mathcal{T}_A + (1-\omega)\ \mathcal{D}_A \tag{5}$$

Cette mise à jour progressive permet à notre traqueur d'être plus tolérant à des détections erronées ou imprécises et évite de trop nombreux calculs comparativement à DeepSORT [26] et d'autres qui stockent un historique des apparences pour chaque trajectoire. Par ailleurs, si une trajectoire n'est pas associée à une détection, son modèle d'apparence n'est pas réactualisé ; le traqueur utilisera alors sa dernière signature connue et de confiance.

Maintien des trajectoires actives non associées. Afin de rattraper d'éventuelles non-détections de la part du détecteur, nous réutilisons la carte de similarité obtenue à l'étape de prédiction de la position (*cf.* section 3.3) pour déterminer si une trajectoire non associée doit être maintenue active ou non. Ainsi, si la valeur maximum de sa carte de prédiction est suffisante (*i.e.* supérieure à σ), sa position est recentrée sur ce point et son score de confiance mis à jour. Néanmoins, une trajectoire maintenue active sur plusieurs images consécutives traduit une incertitude forte de la part du détecteur : cette information est intégrée à la confiance de la trajectoire au travers d'une pénalité appliquée en cas de maintien en activité. Lorsque le score de la trajectoire est trop faible, elle devient inactive. Cette stratégie permet ainsi de compenser les manques du détecteur.

4 Évaluations et discussion

Nous comparons ici notre approche aux autres méthodes de suivi multi-cibles en ligne du MOT Challenge et montrons des performances à l'état de l'art. Nous présentons également l'apport de cet article *via* une étude par ablations.

4.1 Dataset et métriques

MOT Challenge. La communauté vision par ordinateur se compare régulièrement sur ce *challenge* qui met à disposition des scènes complexes et variées, dont la nature (densité, taille des cibles, mouvement caméra, etc.) varie grandement. Le MOT Challenge propose notamment un ensemble de 7 séquences de test dans le MOT17, dans lesquelles l'objectif est de suivre des piétons en mouvement. Comme énoncé en section 2.2, nous nous comparons aux approches utilisant un détecteur privé.

Métriques d'évaluation. Les résultats du MOT Challenge sont évalués selon des métriques permettant de jauger la qualité du système traqueur développé :

1. des métriques CLEARMOT [2] : faux positifs FP, faux négatifs FN, changements d'identité IDS,

fragmentations (FM), combinées dans la métrique MOTA (*Multiple Object Tracking Accuracy*) ;

2. des métriques d'identité [19] : faux positifs d'identité IDFP, faux négatifs d'identité (IDFN), combinées dans la métrique IDF1 (*Identity F1 score*).

Si le MOTA est une métrique largement usitée par la communauté, elle a plutôt tendance à mesurer la couverture globale des trajectoires à suivre et donc – indirectement – la qualité de la partie détection du système. A l'inverse, l'IDF1 est plutôt une métrique permettant de mesurer la qualité du suivi et du maintien des trajectoires identifiées.

4.2 Détails d'implémentation

Toutes les expérimentations sont réalisées à l'aide d'un ordinateur équipé d'un processeur Intel(R) Core(TM) i7-8700 CPU et d'une carte graphique Nvidia RTX 2080 Ti.

Déplacement et association des trajectoires/détections. Le déplacement d'une trajectoire active entre deux instants image étant faible, nous pouvons diminuer les calculs en considérant une zone de recherche égale à la taille de la boîte, celle-ci restant suffisamment grande pour exploiter notre approche multi-hypothèses. Pour le cas des trajectoires inactives, nous fixons $\xi = \frac{1}{7}$ dans l'équation (2). Dans l'équation 3 qui définit le coût d'association d'une trajectoire avec une détection, $\lambda = 0.7$.

Gestion des trajectoires. Pour la mise à jour du score de confiance d'une trajectoire (*cf.* équation (4)), nous fixons $\alpha = 0.2$, $\beta = 0.4$, $\gamma = 0.4$ et $\rho = 0.98$. Dans l'équation (5) de mise à jour du vecteur d'apparence, $\omega = 0.9$. Par ailleurs, une trajectoire active non-associée est maintenue active seulement si son score de confiance $\mathcal{T}_c > 0.4$ et si $\texttt{Sim}_{\mathcal{T},\mathcal{T}} > 0.9$ (*cf.* équation (4)). Enfin, la durée d'inactivité maximale d'une trajectoire est fixée à $n = 30$.

Compensation du mouvement caméra. Afin de rester robuste y compris en contexte caméra mobile, nous estimons son mouvement à l'aide de l'algorithme *Enhanced Correlation Coefficient (ECC)* [6] et le compensons en déplaçant les trajectoires dans la même direction. Cela nous permet ainsi de considérer les hyperparamètres précédents comme génériques, en contexte caméra fixe ou mobile.

4.3 Étude par ablations

TABLE 1 – Étude par ablations réalisée sur la base d'entraînement du MOT17 montrant l'apport de chaque stratégie.

	(A)		(B)	(C)					
RV	**ReID**		**Main.**	**MH**	**IDFP**	**IDFN**	**IDS**	**FM**	**MOTA**	**IDF1**
✓					12725	25182	1829	2042	83.2	82.1
✓	✓				-2414	-2330	-1043	-153	+0.9	+2.3
✓	✓		✓		-1076	-4526	-1121	-592	+1.6	+2.9
✓	✓		✓	✓	**-1772**	**-5222**	**-1115**	**-580**	**+1.6**	**+3.6**

L'étude par ablations présentée dans la table 1 a deux objectifs : démontrer l'apport de chaque contribution détaillée en section 3 et trouver la meilleure configuration pour le traqueur. Partant d'une configuration de base intégrant un recalage visuel (RV), cette étude traite notamment de l'apport du score de réidentification dans l'étape d'association (A), de celui de la stratégie de maintien des trajectoires (B), et enfin de celui de la stratégie de prédiction multi-hypothèses (C).

Prise en compte de la réidentification (A). En plus de la valeur de superposition $\texttt{IoU}_{\mathcal{D},\mathcal{T}}$, considérer le niveau de similarité $\texttt{Sim}_{\mathcal{D},\mathcal{T}}$ amène des gains importants. Ce score permet de n'associer une trajectoire qu'à une détection dont les signatures visuelles sont proches. Nous supprimons ainsi 57% des changements d'identité (IDS) par rapport à la configuration sans ce critère, et apportons une amélioration sur les métriques d'identité (-19% IDFP, -9% IDFN), de même qu'en MOTA (+0.9) et en IDF1 (+2.3).

Maintien des trajectoires (B). La mise en place d'une stratégie de maintien en activité des trajectoires amène une avancée notable sur les résultats. Cette composante permet d'une part d'échanger les identités moins souvent (-61% IDS), mais aussi de mieux maintenir les trajectoires dans le temps (-29% FM). Bien que cet indicateur ne soit pas pris en compte dans les scores d'évaluation, elle traduit la capacité de notre traqueur à ne pas diviser les trajectoires dans le temps. Aussi, elle permet de grandement améliorer les métriques d'identité, en supprimant 8% IDFP et 18% IDFN. Outre l'augmentation importante en IDF1 (+2.9), ces bons résultats se traduisent par des gains importants en MOTA (+1.6) par la compensation d'erreurs de détection.

Stratégie multi-hypothèses et approche globale (C). Là où une stratégie mono-hypothèse filtre certains déplacements candidats pour n'en garder qu'un, l'utilisation de l'intercorrélation permet une stratégie multi-hypothèses qui garde l'information contextuelle de la trajectoire jusqu'à la phase d'association. Autrement dit, elle permet d'associer des couples trajectoire/détection qui auraient *a priori* été écartés en contexte mono-hypothèse.

TABLE 2 – Comparaison de la configuration retenue pour notre approche avec FairMOT sur la base d'entraînement.

	MOTA	**IDF1**
FairMOT	83.8	81.9
C2DT	**84.8 (+1.0)**	**85.7 (+3.8)**

Apport global de la méthode. Ces trois contributions apportent des gains notables sur les résultats finaux. Par rapport au traqueur initial, nous améliorons chacun des critères étudiés. Cela se traduit, au final, par un gain de 1.6 en MOTA et de 3.6 en IDF1. Ainsi, pour un détecteur identique, nos contributions amènent un gain substantiel en comparaison avec les résultats de départ de FairMOT [31] : nous gagnons 1.0 en MOTA et 3.8 en IDF1 (*cf.* table 2). Par ailleurs, il est logique que le gain en MOTA soit moins important, cette métrique étant davantage impactée par le détecteur. Les gains apportés en IDF1, métrique permettant de juger la qualité du traqueur qui constitue notre apport méthodologique, témoignent de l'apport des différentes contributions de cet article.

TABLE 3 – Comparaison des soumissions en ligne avec détections privées sur le jeu de données de test du MOT17 [18]. Les méthodes suivies d'une astérisque (*) n'ont pas encore été publiées dans la littérature.

Class.	Année	Conf.	Méthode	IDF1 ↑	MOTA ↑	MT ↑	ML ↓	FP ↓	FN ↓	IDS ↓	FM ↓
1	2021	arXiv	ByteTrack [30]*	77.3	80.3	1254	342	25491	83721	2196	2277
2	2022	WACV	MAA [22]	75.9	79.4	1356	282	37320	77661	1452	2202
3	2021	ICCV	GRTU [24]	75	74.9	1170	444	32007	107616	1812	1824
4	2021	arXiv	RTv1 [29]*	74.7	73.8	981	546	27999	118623	1374	2166
5	**2022**		**C2DT**	**73.7**	**73.7**	**1191**	**420**	**47406**	**99015**	**2202**	**3066**
7	2021	AVSS	CrowdTrack [23]	73.6	75.6	1095	288	25950	109101	2544	11343
8	2021	CVPR	TLR [25]	73.6	76.5	1122	300	29808	99510	3369	6063
9	2020	arXiv	CSTrack [15]*	72.6	74.9	978	411	23847	114303	3567	7668
10	**2021**	**IJCV**	**FairMOT [31]**	**72.3**	**73.7**	**1017**	**408**	**27507**	**117477**	**3303**	**8073**

4.4 Évaluations et étude comparative

La table 3 met en perspective différentes approches de suivi multi-cibles de la littérature sur les résultats du *challenge* MOT17 avec détections privées. Afin d'établir une comparaison claire et objective, seules les approches en ligne sont mentionnées dans le tableau. Si certaines approches offrent de meilleurs résultats, il faut néanmoins dissocier l'apport du détecteur utilisé de celui du traqueur développé. Ainsi, ByteTrack [30] utilise par exemple le détecteur récent YOLOX [8], aujourd'hui parmi les meilleurs (en particulier, il atteint 51.5% d'AP sur le dataset COCO [16] quand CenterNet [5], le détecteur utilisé par FairMOT que nous reprenons, n'obtient que 34.7%). L'utilisation de détecteurs différents rend difficile la comparaison des traqueurs présentés tant leur impact sur les performances est important. Toutefois, nous observons que, **avec un détecteur identique à FairMOT, nous amenons un gain notable de 1.4 en IDF1**. De ce fait, en considérant deux approches basées sur un même détecteur, nos contributions apportent un meilleur suivi des cibles dans le temps, d'une part en maintenant davantage chacune des trajectoires (3066 FM *vs.* 8073 pour FairMOT [31], soit 62% de moins) et d'autre part en permettant de grandement réduire les changements d'identités (2202 IDS *vs.* 3303, soit 33% de moins).

La figure 3 montre des exemples qualitatifs de notre approche. La première ligne témoigne de la robustesse du traqueur, y compris dans des situations denses et avec des personnes aux apparences similaires. De la même façon, notre stratégie de maintien des trajectoires actives se révèle utile lors d'absences de détections (seconde ligne, partie gauche). Néanmoins, lorsque le détecteur ne parvient pas à identifier une cible pendant une trop longue période (seconde ligne, partie droite), nous finissons par détruire sa trajectoire. Un détecteur plus performant permettrait de conserver cette trajectoire active, et donc d'éviter un changement d'identité *a posteriori*.

FIGURE 3 – Exemples qualitatifs de notre approche. Une boîte colorée épaisse représente une trajectoire active et associée à une détection ; une boîte bleue fine symbolise le maintien de cette trajectoire active non associée à une détection. La première ligne et la partie gauche de la seconde illustrent la qualité de notre traqueur dans des situations denses et avec des erreurs de détection. La partie droite de la seconde ligne montre la perte d'une trajectoire à cause d'un trop grand nombre d'erreurs de détection.

5 Conclusion et perspectives

Dans cet article, nous mettons en avant un couple détecteur-traqueur qui s'appuie sur un unique réseau de neurones servant au recalage visuel des trajectoires et à leur réidentification. Nous proposons une approche originale multi-hypothèses, basée sur l'intercorrélation par apprentissage profond et ne nécessitant pas d'apprentissage supplémentaire pour réidentifier les cibles d'une image à l'autre, de même qu'une stratégie de gestion et de maintien des trajectoires. Nous montrons des résultats à l'état de l'art améliorant la qualité du suivi des cibles et le maintien de leur identité dans le temps. Bien que l'approche présentée amène une amélioration notable sur le suivi multi-cibles, la qualité d'un traqueur repose en partie sur celle des détections en amont. Coupler un détecteur plus performant et entraîné *via* une méthode inspirée de FairMOT à notre traqueur permettrait probablement d'accentuer encore davantage les gains. Il serait également intéressant d'étudier l'apport de l'utilisation de couches d'attention ([4]) en remplacement de l'opération d'intercorrélation.

Remerciements

Je remercie l'ANRT pour son support financier dans le cadre de ma convention de thèse CIFRE chez ACTIA Automotive et au LAAS-CNRS.

Références

[1] P. Bergmann et al., Tracking without bells and whistles, *IEEE Int. Conf. on Computer Vision (ICCV)*, 2019.

[2] K. Bernardin et al., Evaluating multiple object tracking performance : the clear mot metrics, *EURASIP Journal on Image and Video Processing*, Vol. 2008, pp. 1-10, 2008.

[3] A. Bewley et al., Simple online and realtime tracking, *IEEE Int. Conf. on Image Processing (ICIP)*, 2016.

[4] X. Chen et al., Transformer tracking, *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[5] K. Duan et al., Centernet : Keypoint triplets for object detection, *IEEE Int. Conf. on Computer Vision (ICCV)*, 2019.

[6] G. Evangelidis et al., Parametric image alignment using enhanced correlation coefficient maximization, *IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI)*, Vol. 30, pp. 1858–1865, 2008.

[7] W. Feng et al., Multi-object tracking with multiple cues and switcher-aware classification, *arXiv :1901.06129*, 2019.

[8] Z. Ge et al., Yolox : Exceeding yolo series in 2021, *arXiv :2107.08430*, 2021.

[9] D. Held et al., Learning to track at 100 fps with deep regression networks, *European Conf. on Computer Vision (ECCV)*, 2016.

[10] R. E. Kalman, A New Approach to Linear Filtering and Prediction Problems, *Trans. of the ASME–Journal of Basic Engineering*, Vol. 82, pp. 35-45, 1960.

[11] M. Kristan et al., The Seventh Visual Object Tracking VOT2019 Challenge Results, *IEEE Int. Conf. on Computer Vision (ICCV)*, 2019.

[12] F. Schroff et al., Facenet : A unified embedding for face recognition and clustering, *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[13] C. Labit-Bonis et al., Compact and discriminative multi-object tracking with siamese CNNs, *IEEE Int. Conf. on Pattern Recognition (ICPR)*, 2021.

[14] B. Li et al., High performance visual tracking with siamese region proposal network, *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[15] C. Liang et al., Rethinking the competition between detection and ReID in Multi-Object Tracking, *arXiv :2010.12138*, 2020.

[16] T.-Y. Lin et al., Microsoft coco : Common objects in context, *European Conf. on Computer Vision (ECCV)*, 2014.

[17] A. A. Mekonnen et al., Comparative evaluations of selected tracking-by-detection approaches, *IEEE Trans. on Circuits and Systems for Video Technology (TCSVT)*, Vol. 24, pp. 996-1010, 2019.

[18] A. Milan et al., MOT16 : a benchmark for multi-object tracking, *arXiv :1603.00831*, 2016.

[19] E. Ristani et al., Performance measures and a data set for multi-target, multi-camera tracking, *European Conf. on Computer Vision (ECCV)*, 2016.

[20] R. Sanchez-Matilla et al., Online multi-target tracking with strong and weak detections, *European Conf. on Computer Vision (ECCV)*, 2016.

[21] Y.-M. Song et al., Online multi-object tracking with GMPHD filter and occlusion group management, *IEEE Access*, 2019.

[22] D. Stadler et al., Modelling Ambiguous Assignments for Multi-Person Tracking in Crowds, *IEEE Winter Conf. on Applications of Computer Vision (WACV) Workshops*, 2022.

[23] D. Stadler et al., On the Performance of Crowd-Specific Detectors in Multi-Pedestrian Tracking, *IEEE Int. Conf. on Advanced Video and Signal Based Surveillance (AVSS)*, 2021.

[24] S. Wang et al., A General Recurrent Tracking Framework Without Real Data, *IEEE Int. Conf. on Computer Vision (ICCV)*, 2021.

[25] Q. Wang et al., Multiple Object Tracking with Correlation Learning, *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[26] N. Wojke et al., Simple Online and Realtime Tracking with deep association metric, *IEEE Int. Conf. on Image Processing (ICIP)*, 2017.

[27] Y. Xu et al., SiamFC++ : Towards robust and accurate visual tracking with target estimation guidelines, *AAAI Conf. on Artificial Intelligence*, Vol. 34, pp. 12549-12556, 2020.

[28] A. Yilmaz et al., Object tracking : a survey, *ACM Computing Survey*, 2006.

[29] E. Yu et al., RelationTrack : Relation-aware Multiple Object Tracking with Decoupled Representation, *arXiv :2105.04322*, 2021.

[30] Y. Zhang et al., ByteTrack : Multi-Object Tracking by Associating Every Detection Box, *arXiv :2110.06864*, 2021.

[31] Y. Zhang et al., Fairmot : On the fairness of detection and re-identification in multiple object tracking, *IEEE Int. Journal of Computer Vision (IJCV)*, Vol. 129, pp. 3069-3087, 2021.