Titre: Title:	Multiple Object Tracking in Urban Traffic Scenes
Auteur: Author:	Hui Lee Ooi
Date:	2021
Type:	Mémoire ou thèse / Dissertation or Thesis
Référence: Citation:	Ooi, H. L. (2021). Multiple Object Tracking in Urban Traffic Scenes [Ph.D. thesis, Polytechnique Montréal]. PolyPublie. https://publications.polymtl.ca/5592/

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Multiple Object Tracking in Urban Traffic Scenes

HUI LEE OOI
Département de génie informatique et génie logiciel

Thèse présentée en vue de l'obtention du diplôme de *Philosophiæ Doctor*
Génie informatique

Janvier 2021

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Cette thèse intitulée :

Multiple Object Tracking in Urban Traffic Scenes

présentée par **Hui Lee OOI**
en vue de l'obtention du diplôme de *Philosophiæ Doctor*
a été dûment acceptée par le jury d'examen constitué de :

Giovanni BELTRAME, président
Guillaume-Alexandre BILODEAU, membre et directeur de recherche
Nicolas SAUNIER, membre et codirecteur de recherche
Lama SÉOUD, membre
Robert LAGANIÈRE, membre externe

DEDICATION

To those who are dear to me

ACKNOWLEDGEMENTS

The past five years was a journey of discovery, development and realization, both in the professional and personal sense for me. Now that I am on the cusp of wrapping up this chapter of my life, I would like to express my appreciation to all the individuals that have inspired me and enriched my life in different ways and degrees through the ride.

First and foremost, I would like to express my deepest gratitude to my supervisor Prof. Guillaume-Alexandre Bilodeau for his invaluable guidance, patience, encouragement and motivation during this time. I am also deeply thankful for my co-supervisor Prof. Nicolas Saunier for his insight and inputs that are immensely helpful during the completion of my research project. I am grateful for their mentorship, feedback and advice that keep me in track in my research progress.

Also, many thanks to my fellow labmates that I have met and worked with in LITIV lab. Special thanks to Pierre-Luc for his advice and technical help when I first started the journey as well as Hughes and David-Alexandre for their contribution and help when I was working on my project. Despite not being able to list each and everyone of them here, their presence had made my foray into the world of academia more memorable, enjoyable and fun.

Next, I am thankful for Fonds de Recherche du Quebec-Nature et Technologies(FRQ-NT) (Grant: 2016-PR- 189250) and Polytechnique Montréal PhD Fellowship for the funding that supported this research project.

On a more personal note, I am thankful for my family for being supportive of my decision of embarking on this adventure. I thank my friends for the words of encouragement and well wishes for my endeavour. Also, thanks to my fellow hiking buddies that do a lot more than just hiking, letting me de-stressing as I navigate through my academic pursuits.

Finally, I would like to thank the members of jury for their acceptance of reviewing this thesis.

RÉSUMÉ

Le suivi multiobjets (MOT) est un domaine très étudié qui a évolué et changé beaucoup durant les années grâce à ses plusieurs applications potentielles pour améliorer notre qualité de vie. Dans notre projet de recherche, spécifiquement, nous sommes intéressés par le MOT dans les scènes de trafic urbain pour extraire précisément les trajectoires des usagers de la route, afin d'améliorer les systèmes de circulation routière desquels nous bénéficions tous.

Notre première contribution est l'introduction d'informations sur les étiquettes de classe dans l'ensemble des caractéristiques qui décrivent les objets pour les associer sur différents trames, afin de bien capturer leur mouvement sous forme de trajectoires dans un environnement réel. Nous capitalisons sur les informations provenant d'un détecteur basé sur l'apprentissage profond qui est utilisé pour l'extraction des objets d'intérêt avant la procédure de suivi, car nous avons été intrigués par leurs popularités croissantes et les bonnes performances qu'ils obtiennent. Cependant, malgré leur potentiel prometteur dans la littérature, nous avons constaté que les résultats étaient décevants dans nos expériences. La qualité des détections, telle que postulée, affecte grandement la qualité des trajectoires finales. Néanmoins, nous avons observé que les informations des étiquettes de classe, ainsi que son score de confiance, sont très utiles pour notre application, où il y a un nombre élevé de variabilité pour les types d'usagers de la route.

Ensuite, nous avons concentré nos efforts sur la fusion des entrées de deux sources différentes afin d'obtenir un ensemble d'objets en entrée avec un niveau de précision satisfaisant pour procéder à l'étape de suivi. À ce stade, nous avons travaillé sur l'intégration des boîtes englobantes à partir d'un détecteur multi-classes par apprentissage et d'une méthode basée sur la soustraction d'arrière-plan pour résoudre les problèmes tels que la fragmentation et les représentations redondantes du même objet. Nous avons décidé d'utiliser une meilleure formulation pour la distance spatiale entre les objets utilisée pendant le processus d'association des données. Dans notre méthode de suivi proposée, le paradigme de suivi est également amélioré avec une évaluation de la qualité de la prédiction, où la prédiction est utilisée dans les trajectoires lorsque des occlusions et des interactions entre les objets causent des difficultés lors des correspondances dans certaines trames.

Compte tenu des défis que nous avons rencontrés au cours de l'avancement de la recherche, nous avons été motivés à réaliser une étude systématique et quantitative pour comparer et évaluer les effets de la détection supervisée et de la détection non supervisée sur nos performances de suivi. Notre méthode de suivi proposée inclut désormais des vecteurs de

description de ReID dans l'association des données en raison de leur efficacité rapportée dans les publications récentes. Nous avons adopté pour un détecteur par apprentissage profond plus récent et plus puissant et une version modifiée d'une méthode de soustraction d'arrière-plan pour une comparaison équitable des entrées de détection sous la forme de boîtes englobantes. Pour cette expérience, nous avons travaillé sur les ensembles de données UA-Detrac et Urban Tracker. Les résultats montrent que le détecteur par apprentissage est capable de fournir de meilleures entrées pour le suivi, si un grand ensemble de données est disponible pour permettre un entraînement suffisant. Par contre, il ne fonctionne pas très bien sur un ensemble de données de plus petite taille avec une grande variabilité des scènes. À l'inverse, les entrées de la méthode basée sur la soustraction d'arrière-plan ne sont pas très robustes dans le cas d'un ensemble de données contenant du bruit en raison des mouvements de la caméra et des perturbations environnementales. Nous avons observé que la soustraction d'arrière-plan donne de meilleurs résultats lorsque la taille de la base de données est limitée, car elle est capable d'identifier des objets inattendus dans la trames sans entraînement intensif préalable.

En somme, notre analyse montre l'importance d'une bonne extraction des objets en MOT, car les erreurs de détection se propagent souvent dans le pipeline de suivi et affectent sévèrement les performances de la méthode de suivi. Nous avons également étudié les types de détecteurs (supervisés et non supervisés) avec différents ensembles de données et nous concluons que différents ensembles de données nécessitent différentes manières d'extraire efficacement les objets pour l'application du MOT. Finalement, en dehors de cela, nous avons également démontré que la stratégie que nous proposons d'utiliser les informations d'étiquettes de classe et l'évaluation de la qualité des prédictions améliorent significativement les performances de suivi.

ABSTRACT

Multiple object tracking (MOT) is an intensively researched area that have evolved and undergone much innovation throughout the years due to its potential in a lot of applications to improve our quality of life. In our research project, specifically, we are interested in applying MOT in urban traffic scenes to portray an accurate representation of the road user trajectories for the eventual improvements of road traffic systems that affect people from all walks of life.

Our first contribution is the introduction of class label information as part of the features that describe the targets and for associating them across frames to capture their motion into trajectories in real environment. We capitalize on that information from a deep learning detector that is used for extraction of objects of interest prior to the tracking procedure, since we were intrigued by their growing popularity and reported good performances. However, despite their promising potential in the literature, we found that the results were disappointing in our experiments. The quality of extracted input, as postulated, critically affects the quality of the final trajectories obtained as tracking output. Nevertheless, we observed that the class label information, along with its confidence score, is invaluable for our application of urban traffic settings where there are a high number of variability in terms of types of road users.

Next, we focused our effort on fusing inputs from two different sources in order to obtain a set of objects with a satisfactory level of accuracy to proceed with the tracking stage. At this point, we worked on the integration of the bounding boxes from a learned multi-class object detector and a background subtraction-based method to resolve issues, such as fragmentation and redundant representations of the same object. We decided to employ a better formulation to account for the spatial distance between objects that is used during the data association process. In our proposed tracker, the tracking paradigm is also improved with an evaluation of prediction quality, where a prediction is used as a trajectory element when occlusions and interactions among objects cause difficulty in matching them during some frames in the video.

Given the challenges that we have encountered during the research progress, we were motivated to perform a systematic and quantitative study to compare and evaluate the effects of supervised detections and unsupervised detections on our tracking performance. Our proposed tracker now includes ReID features in data association due its effectiveness reported in some of the more recent literature. We adopted a newer and more powerful deep learning

detector and a modified version of a state-of-the-art background subtraction method for a fair comparisons of inputs in the form of bounding boxes. For this part of the thesis, we also worked on the UA-Detrac dataset in addition to the Urban Tracker dataset that we have used previously. The results have shown that while the learned object detector is capable of giving better inputs that aid tracking on large-scale dataset that facilitate sufficient training, it is not performing very well on a dataset that is smaller in size but with high variability of scenes. Conversely, inputs from background subtraction-based method might not be very competitive in datasets that contain noisy elements due to camera movements and environmental perturbations. However, we have observed that it actually gives better results when handling urban scenes with limited size as it is capable of identifying unexpected objects in the scene without prior intensive training.

In summary, our analysis has shown the importance of good extraction of objects in MOT, since the errors in detecting them often propagate down the pipeline and severely affect the performance of the tracker. We also investigated the types of detectors (supervised and unsupervised) in different datasets and concluded that different datasets require different ways of approaching the problem when it comes to extract the objects effectively for the application of MOT. Finally, apart from that, we have also demonstrated that our proposed strategy of using class label information and prediction evaluation has significantly improved the tracking performance.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ACRONYMS

CNN	Convolutional Neural Network
FPN	Feature Pyramid Network
FSM	Finite State Machine
GRU	Gated Recurrent Unit
IOU	Intersection over Union
KCF	kernelized correlation filter
LSTM	long short-term memory
MOT	multiple object tracking
MOTA	Multiple Object Tracking Accuracy
MOTP	Multiple Object Tracking Precision
MHT	Multiple Hypotheses Tracking
ReID	Re-Identification
RNN	Recurrent Neural Network
SVM	Support Vector Machine
SOT	single object tracker
SOTA	state-of-the-art
VOT	visual object tracker

CHAPTER 1 INTRODUCTION

1.1 Background

According to a report by the United Nations [5], the world population is booming at a rapid pace and is expected to reach 9.7 billion by the year 2050. Urbanization is spurred by this trend and massive migration to large cities is continuing because of better economic opportunities. The rapid development and inevitable population growth of these cities and urban areas, however, are contributing to traffic congestion and other mobility problems.

The growing traffic demand far exceeds the road network capability during peak hours, causing the roads to be more congested and increasing queuing. Such deterioration of traffic conditions have caused much inconvenience to the road users and to public transportation that uses those same infrastructure, as well as induced massive inefficiency in the traffic network. Studies from [6] and [7] have shown that traffic congestion severely impacts the environment with higher noxious gases emission, and a greater carbon footprint. The air pollution critically affects the physical health of citizens, resulting in more reported cases of chronic respiratory diseases in recent years [8]. Various studies in the past [9, 10] have revealed traffic congestion to be a source of stress that affects the driving behavior on the roads as well. In addition, prolonged traffic congestion will leave the drivers in a state of fatigue, causing them to be more prone to reckless driving and thus potentially leading to a higher rate of accidents. The time of each individual getting stuck in the traffic is wasted instead of being spent on important and precious moments in life such as family bonding and engaging in activities that promote self-improvement physically, mentally and spiritually. It is apparent that this phenomenon of traffic congestion causes deterioration in the quality of life for all citizens in varying degree.

Fortunately, some possible solutions to these problems may be found in Intelligent Transportation System (ITS). It is envisioned that better data collection will improve analysis and management of the traffic in real-time. The control system of an ITS can improve traffic by matching the demand with transportation supply. Traffic control requires good real-time data on all the road users in the traffic to fully interpret and analyze the traffic states accurately and efficiently. Consequently, the task of tracking all road users in traffic scenes with constantly changing conditions is of utmost importance.

Currently, transportation departments have many cameras installed to manually monitor the traffic and to detect unexpected incidents. However, many of them are from older technology

with low quality images and some are even still in analog format. These videos are recorded with low frame rates and high compression rates, making it difficult for automatic processing. The current progress in the imaging technology, on the other hand, has led to the availability of high performance cameras with high image quality and low price. Video surveillance systems with these cameras provide an attractive and economical platform for data collection and automated traffic monitoring. The footage of the various road users from the cameras could be processed and analyzed, where the resulting information may be used for the purpose of traffic control, safety analysis and behavioral analysis.

It is envisioned that the evolving technological approaches in recent years and in the near future will be able to fully capitalize on these higher quality image data to produce a more thorough and efficient analysis as well as processing of information on the traffic conditions, even in complicated urban traffic scenes with various variables to eventually improve the traffic system in the long term.

1.2 Problem Statement

Given that multiple object tracking (MOT) holds much potential in collecting invaluable traffic information such as vehicle counting and trajectories in urban traffic scenes, we propose it as part of the potential solutions that can be integrated in an ITS for improved traffic management. Nevertheless, despite its growing popularity and attention in the field of transportation, there are still much to be addressed and resolved [2, 11].

In the context of this thesis, we are interested in solving the MOT problem in urban traffic scenes that involve a number of moving objects under good visibility condition with an online approach, as defined in Section 2.1. The road users are the objects of interest to track in the MOT task and they include pedestrians, cars, bicycles and heavy vehicles. The term road users may be used interchangeably in the rest of the thesis with the terms “objects” or “targets”. The ideal output for this task is to produce trajectories that represent the movements of targets in the traffic scene as they move in the actual traffic environments in real-time.

However, in the process of detecting the moving road users for tracking, it is common to encounter object deformation or poor object localization due to occlusions in crowded scenes. Erroneous or mistaken detections resulting from occlusion may propagate the inaccuracies to the tracking stage. The problem of occlusion (see Figure 1.1) greatly impacts the trajectories as the target is completely or partially obstructed and thereby is no longer visible to the road user detector. It is difficult to detect such objects due to their variability in appearance

when occluded. In some case, objects may simply be not detected, or there might be false detections due to similarities with the environment or non-targets in the background.

Figure 1.1 Examples of occlusions in urban traffic scenes.

Additionally, depending on the type of object detectors, there is the problem of distinguishing between a split and a fragmented object. Splitting objects occurs when two different targets in close proximity move together before moving to different directions, which could be a common occurrence at traffic intersections. Fragmentation, meanwhile, is a problem when there are some pixels in the target that is not detected as the foreground region, causing the object to appear to be broken into pieces (see Figure 1.2). Different works have attempted to solve this issue with varying degrees of success [12–16].

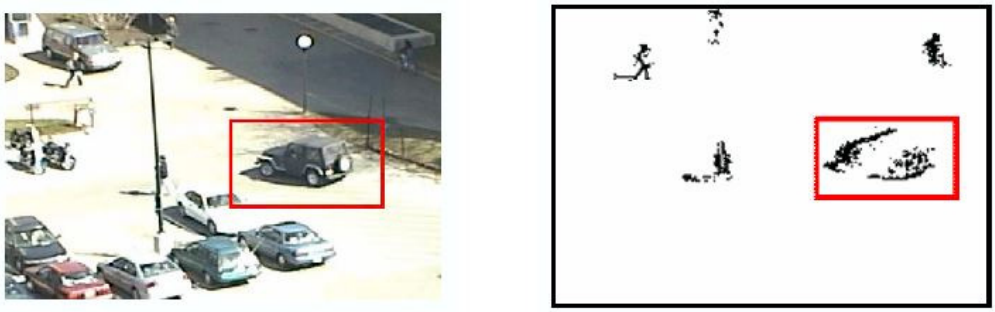


Figure 1.2 Example of fragmentation of target [1]

Another common problem is the variation of illumination and appearance in the video. An effective tracking algorithm therefore has to take into consideration the illumination or appearance change by updating the tracker. Due to the presence of various targets in MOT, the interaction between different objects in the scenes may complicate the update process even

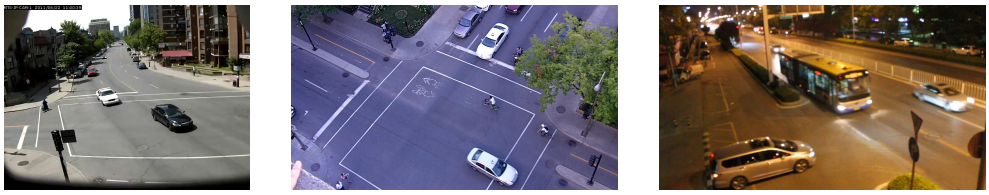
more over time. That is, an object description should not be updated when an object is partly hidden, because that representation will not be the real appearance of the object. Hence, an efficient tracker requires the capability of frequent updates of description for each road user to avoid propagation of inaccurate representation that can cause failed tracking. The object description should also capture the appearance with a good precision to avoid confusion and tracking errors (see Figure 1.3). A single feature like color is not always sufficient.

Figure 1.3 An example of different objects with similar color appearance.

Research on MOT for transportation applications has been in large part focused in highway settings [17, 18], where there is less variety in the types of road users and more homogeneous movements of road users and therefore the task is less challenging to solve. On the other hand, this project aims to propose a MOT approach to be used in cities and urban areas with signalized intersections as shown in Figure 1.4. The potentially complicated interactions and trajectories of the different road users, including stopped vehicles, pedestrians and cyclists, make the task even more challenging and difficult than the conventional constraints faced by MOT solutions. The change of orientation and appearance of the different objects due to varying pose and rotation as they move in the scenes remains one of the major challenges to be addressed. Dealing with detection errors is another major challenge in urban scenes.

1.3 Definition of MOT

The term “objects” in MOT, also known as targets, refers to image areas from a video frame that are distinct from their surroundings and pose certain significance such that we are interested in their movement in the subsequent frames. Objects in MOT are attributed a unique identity (ID) to distinguish them throughout a video sequence. In urban traffic scenes, the objects of interest would be the different road users such as pedestrians and different



(a) (b) (c)

Figure 1.4 Examples of selected traffic scenes of urban traffic.

vehicles. By definition, detection is the localization process performed on the image frames to extract image area of the targets. Detection responses, often referred to as "Detections", are the end result from the detection process, in which extraction of specific targets such as vehicles are obtained either with a supervised or an unsupervised method. Tracking is the localization of the multiple objects across multiple frames. A trajectory is a sequence of positions over time that represents each object and is the main output from the tracking process. One trajectory consists of multiple detection responses of a unique target. It should not be confused with a tracklet that is used in some works in the literature [19], which is the intermediate stage between detection responses and trajectories. A tracklet is obtained by linking confident detection responses that are believed to belong to the same target. It is shorter in time span compared to a trajectory and is a subset of the whole trajectory of the object. The final trajectories could eventually be constructed by progressively linking tracklets, as shown by Figure 1.5.

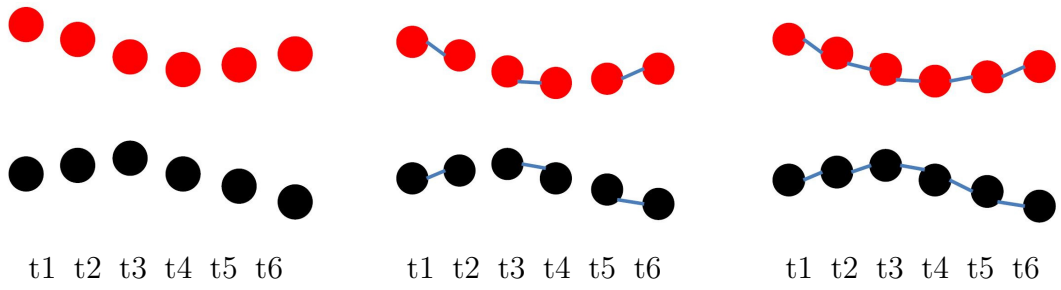


Figure 1.5 Examples of detection response (left), tracklets (center) and trajectories (right) of two separate targets across six frames from $t1$ to $t6$ [2](version 3)

Data association is the matching process for the different detection responses for each of the targets across frames. In short, the ultimate objective of MOT task is to produce from object detection the trajectories of all targets with a high level of precision.

1.4 Research Objectives

Given that there are a number of existing problems with regards to MOT in urban traffic settings, specifically missing detections, occurrence of occlusion and similarity of appearance among targets, this research work intends to address a number of existing problems with regards to MOT. The main objective of this project is to propose and develop a MOT solution tailored for the application in urban traffic scenes that contain varying types of road users simultaneously, taking into consideration the interaction among the targets as well as with the background.

This project aims to achieve three fundamental research objectives, specifically:

1. To study and compare object detection methods to extract the different targets (road users) from the background of video frames, accounting for the common issues in MOT, such as occlusion, fragmentation, false and missing detections.

2. To define and propose suitable features or combination of features to represent the targets that address variations in illumination and appearance to better distinguish the multiple targets from one another. The targets would be analyzed according to these specific characteristics for the tracking task.

3. To propose a data association strategy that performs satisfactory matching of targets across frames and is capable of handling appearance variations, possible fragmentation and objects with similar appearance, forming trajectories that represent targets with high accuracy.

1.5 Contributions

The main contributions of this thesis are presented in the form of published papers (Chapter 4, Chapter 5 and Chapter 6):

- In the first paper (Chapter 4) that covers objectives 1 to 3, we proposed a MOT strategy that incorporated a modern object detector. We have also proposed the novel use of class label as a component of combination of features that is used in the data association process. We evaluated the method on the Urban Tracker dataset [20].

- In the second paper (Chapter 5) that covers objectives 1 and 3, we proposed to fuse the detections from both a supervised and an unsupervised approach to give better inputs for our tracker. We have also improved our tracking framework by incorporating the quality evaluation of track prediction for better final tracking outputs.

- In the third paper (Chapter 6) that covers objectives 1 and 2, we performed a thorough evaluation of a supervised detector and an unsupervised detector on the tracking performances on Urban Tracker dataset [20] that we have experimented before as well as on the UA-Detrac dataset [3]. A more recent detector is employed with the training and testing performed on the new dataset. We also added ReID features as part of our feature combination that represent the targets during the tracking process.

1.6 Thesis Structure

This thesis is structured as followed: Chapter 2 delves deep into the current works that have been presented and proposed recently in a similar context of MOT. Chapter 3 gives an overall presentation of the process of approaching and solving the MOT problem in this thesis. The subsequent chapters (Chapter 4, Chapter 5 and Chapter 6) are the published articles as a result of the research done in this project. Chapter 7 presents an overall discussion of the works and experiments performed throughout the project, including some initial unpublished attempts at the early stage of the research. Last but not least, Chapter 8 provides the conclusive remarks as well as recommended ideas for future directions of the research in similar context.

CHAPTER 2 LITERATURE REVIEW

The approaches proposed to solve the MOT task have evolved throughout the years. On a broader scale, there are several aspects of the MOT problem that have been targeted for improvement over time. In this chapter, we are going to explore the previous works and the current literature presented on the different strategies that have been used to tackle the problem of MOT. The proposed approaches can often be viewed as modular solutions, focusing on

- Object Extraction (detection)
- Data Association
- Object Description
- Object Prediction

Nonetheless, there are recent trends that have attempted to combine some of these steps simultaneously in their tracker implementation.

2.1 Types of MOT solutions

In terms of implementation, several categories of proposed approaches were used in MOT applications throughout the years, as shown in Figure 2.1.

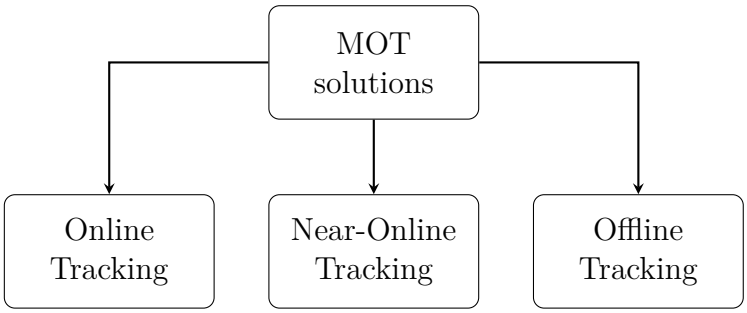


Figure 2.1 Trends of MOT approaches.

Online tracking, also known as sequential tracking [19, 21–25] processes only current and past frames, hence the data association is performed in a step-wise manner with real-time applications. Nonetheless, imprecise tracking can propagate the errors such that it leads to

difficulty in tracking recovery in some cases. On the other hand, offline tracking, also known as batch tracking or global tracking [26–29] utilizes all the frames (past, current and future) in its solution. With additional information from the future frames, offline methods can often produce results with higher accuracy, benefiting from global and optimal approaches, but often at the cost of high computation and memory requirements. As indicated by the name, near-online tracking [30, 31] offers a sweet spot between robust offline methods and efficient online methods, optimizing only a few recent frames at the same time. It does not produce trajectories immediately but does it after certain number of frames, thus avoiding some local optimal errors.

2.2 Extraction of Objects

In order to follow an object, we need to see it first. For initiating the tracking task, the objects are first detected. From the traditional extraction of objects in a frame through digital image processing techniques, the emerging trends in recent works have turned to detection based on convolutional neural network (CNN) for extraction of objects to capitalize on its good performance in the field of object detection. CNN is a class of deep learning networks that have been popular with applications of computer vision due to its ability to process visual information effectively in images. In the following, we will present the object extraction methods that were used in previous works, as well as some highlights of the works to provide context.

[32] proposed a feature based tracker, Traffic Intelligence, which is an open source project that adapted an implementation of the Kanade-Lucas-Tomasi tracker described in [33]. It used grouped features points obtained from sparse optical flow, with consideration for multiple entrance and exit regions, variable trajectories as well as possible feature track disruption. [32] used optical flow to estimate the motion of pixels in an image using a constraint on brightness constancy. The proposed algorithm performed the feature grouping by a graph construction where the vertices represent feature tracks and the edges represent grouping relationship between the tracks. However, since the method is based on optical flow to extract objects, stopping of objects can disrupt tracks and thus cause track fragmentation.

[20] chose to utilize the background subtraction method called ViBe to extract unknown road users from a scene due to missing or false detection that could be given by object detectors at that time. Background subtraction methods learn to separate objects from the background using the color contrast between the pixels in the current frames and the pixels of a reference background frame. They are typically unsupervised methods and can detect any classes of objects. However, objects can be fragmented into parts. In the work of [20],

a modification to the background subtraction approach is performed to handle intermittent object motion in urban traffic scenes, producing objects of interests outputted in the form of blobs. This was to solve the problem of cars being absorbed into the background when they stopped at red lights. Low-level blob matching is first performed in the "low-level" tracking step to form short tracklets (s-tracklets), which are then later assigned to object tracks. This process is applied to deal with object fragmentation typical of background subtraction-based detections.

The tracker proposed by [34] applied background subtraction to extract the moving objects of interest and complemented those with bounding boxes obtained with the VOT (Visual Object Tracker) called Kernelized Correlation Filters (KCF) in urban traffic scenes. A VOT uses appearance to propagate a bounding box in the following frame. In contrast to a MOT method, it is designed to track one object at a time. In this case, the tracking with KCF was initiated with background subtraction when new objects were appearing. Combining bounding boxes from a VOT and background subtraction allowed better tracking consistency during occlusions, as background subtraction cannot separate occluding objects.

[35] proposed a combination of background subtraction and pre-processing steps that involves the edges of targets for the purpose of object extraction. Optical flow is used to handle occurrence of occlusion. The goal of that work was to diminish the inconvenience of background subtraction by separating occluding objects and by merging object fragments. The foreground obtained in blobs from the background subtraction is merged if they are close in proximity and have similar optical flow. If the objects are moving in different directions according to the optical flow, a decision will be taken on the information to construct a new foreground image. This method is capable of eliminating the shadow that came with background subtraction as well as adjusting the size of the original foreground blobs to allow more accurate tracking.

[22] utilized the deep learning detection approach called Faster Region CNN (Faster R-CNN) in its implementation, noting that better detector gives better tracking results in the end. Object detectors learn the appearance in the image of predefined object classes using machine learning techniques. The advantage of these methods is that the objects are detected as a whole, and are not fragmented. The objects can also be separated during partial occlusions. However, these methods cannot detect unexpected objects as they are trained on predefined classes. Given the success of object detectors, [36] used a CNN-based network for the extraction of objects that aids the tracking task. Similarly, [37] compared several detectors, such as Faster R-CNN [38], Recurrent Rolling Convolution (RRC) [39] and TuSimple [40] detectors to extract objects of interest in their MOT method and RRC, a

single stage end-to-end trainable object detection architecture, is reported to give the best performance in the proposed framework.

In the approach by [41], the inputs to the tracker are a pair of RGB detection images with binary body masks from Mask R-CNN [42] in order to focus only on the object's appearance and not the background context. Indeed, methods such as Mask R-CNN detect objects in an image, as well as segment them. If the segmentation is successful, this allows even a better handling of object occlusions since overlapping bounding boxes might give confusing appearance information.

[43] used both a person detector and a body joint detector as their inputs to their proposed tracker, where the joint classifier was trained with the PoseTrack dataset from [44], as an effort to capitalize on the joint information to boost the performance of pedestrian tracking task.

2.3 Data Association

Data association is a major component in MOT and many innovative approaches have been proposed in this direction to improve the performance of MOT. Popular choices for matching the objects across frames are online approaches, such as bipartite graphs solutions that use only past and current frames, as well as offline approaches such as graph networks that uses future frames too. In the following, we will summarize various strategies used for data association. Most methods aim at filtering possible associations to reduce complexity and remove impossible pairings of objects between frames, while favoring the most likely ones.

2.3.1 Using object states and hierarchical data association

First of all, MOT methods often attribute states to object tracks, and states influence the behavior of data association. For example, the tracker by [45] uses a state machine that defines "Initial", "Entry", "Activate", "Occluded" and "Exit" states on the targets. The assignment of target is performed with a greedy search approach, but tracks in the "Exit" state are ignored, and tracks in "Activate" state are associated in priority versus tracks in the "Occluded" state where the appearance of objects is more ambiguous.

UrbanTracker proposed by [20] employs a Finite State Machine (FSM) for data association in MOT to handle blob merging, splitting and fragmenting. The "high-level" tracking involving joining short tracklets (s-tracklets) into final tracks that are assigned in the FSM and a track life cycle (entering, exiting, visible) is represented in this manner. Interpretation of s-tracklets triggers the transition of states. A track stays in "Normal" state if an s-tracklet

with unique source and destination blobs are added without ambiguity. A track that cannot be linked to a s-tracklet transitions into "Lost" state, and it can be moved to "Deleted" state if it is not able to correspond to available s-tracklets consecutively for a certain number of frames. An unassociated blob that can be linked to a track in the "Lost" state will transition the said track into "Normal" state. However, unassociated blobs that are not linked with any active tracks would be put into "Hypothesis" state to take into account the occurrence of noisy and unstable tracks. In such case, s-tracklets that can be added to this track for more than three frames would be moved to "Normal" state, and the ones that cannot will be removed. Evaluation of tracks for object exiting, splitting of tracks and fragmentation are also proposed with keypoints being taken into consideration and with the involvement of the "Hypothesis" state. The data association in this work is more complex because background subtraction can give fragmented objects.

[34] proposed Multiple Kernelized Correlation Filter (MKCF) that utilizes multiple KCF for each of the target for the data association based on the correspondences of the proximity and the internal model of the VOT. Regions of targets were extracted and processed with several morphological operations such as median filtering, closing and hole filling to obtain a list of final candidate object regions. They are later compared with the tracker outputs and different states (tracked, occluded, new, invisible) are defined depending on the overlaps and the previous states. In order to handle occlusion, groups of objects are labeled as being inside a specific group, where the search is performed from redundant KCF trackers among other group members for tracker re-assignment when a member split from the group. Therefore, in this work, data association is only local with respect to the tracked objects.

2.3.2 Using bipartite graphs

Several works in the literature employed the use of bipartite graphs to perform the data association, where the matching of objects is implemented between two frames.

[46] used the Kuhn-Munkres algorithm, also known as the Hungarian algorithm for the association of tracklets and new detections for a global optimal result with a two-stage matching strategy. The matching is initially implemented between detections and tracklets with high tracking quality before moving on to the matching between detections and tracklets with lower tracking quality that are shorter in length. This strategy lowers the possibility of erroneous associations.

In the work by [47], an online divide-and-conquer strategy with Correlation Clustering is used, where global assignment of objects are partitioned in local sub-problems and resolved via selective choice and combination of the best visual features. Their unified structural

learning framework involves the "Divide" step that break into localized association problems and "Conquer" step that selectively combine preferred visual features from an extended Hungarian algorithm-based association scheme. A Latent Structural Support Vector Machine (SVM) framework is applied to combine these two steps and learn the tracker parameters.

[22] proposed the Simple Online and Realtime Tracking (SORT) method that applies the Hungarian algorithm in a minimalisitic and efficient manner of tracking-by-detection, and yet achieves comparable performance with other state-of-the-art (SOTA) methods. Extending the work of [22], [48] used the Hungarian algorithm in a similar manner as well.

[30] proposed the Near-Online Multi-Target Tracking (NOMT) algorithm, where the hypothesis generation and selection scheme for choosing the hypothesis in graphical model are framed as an energy minimization framework. The single target consistency is computed based on the compatibility between the hypothesis and detections from their proposed metric. The Hungarian algorithm is integrated in the work to perform matching and a Kalman filter is used to obtain continuous trajectories from the discrete detection sets.

In the work by [49], pairs of detection-object with minimal structural variation would be associated preferentially and a total probability framed with both appearance and structure cues is proposed. Objects are categorized into matched and unmatched using the Hungarian algorithm after taking into consideration the structural variation cost, followed by the search of the nearest neighbour with intersection over union (IOU) criteria.

In the work by [21], following their previous work in [19], high confidence and low confidence association are introduced to solve the data association problem of MOT. Reliable tracklets with high confidence values have the priority to be locally associated with the detections first, followed by the tracklets with low confidence in global association. The tracklet-detection pairs are assigned by the Hungarian algorithm.

From a probability hypothesis density particle filter framework, [50] exploits both high and low confidence targets from detections. The low confidence detections (weak) are used to support the label propagation whereas the high confidence detections (strong) are used for both label propagation and target initialization. The Hungarian algorithm is used in early association among the predicted states. The early association strategy allows the newly generated particles to inherit the properties of the associated state or to be initialized as a new target.

In the tracker by [51] that deals with multiple targets and multiple cameras, the data association is solved by using correlation clustering that is capable of inferring missing objects by enforcing transitivity. All the pairwise associations are considered and the identities are

jointly optimized. In order to reduce complexity, multi-level reasoning with sliding temporal window is employed: the first level produces tracklets that last for one second, the second level associates the tracklets into single camera trajectories, and the last level subsequently associate them into multi-camera identities.

[52] presented the IOU tracker that leveraged on the good detection results of newer object detectors, using IOU of bounding boxes to fill in the "gaps" that were missed out by the detectors, with the assumption that the frame rate for the video is sufficiently high for such approach. Assignment of detections to the tracks is performed based on the threshold set on their IOU. Despite the simplicity, this tracking-by-detection method was extremely competitive with the SOTA methods of their time. [53] further extended this work with the inclusion of visual information to handle cases of ID switches and fragmentation with the V-IOU tracker. Due to the fragmented nature of the tracks in the original IOU tracker, a gap filling task is further implemented via a VOT, propagating the objects across frames by using the visual information. To account for the false detection that might interfere with the tracking process, the resultant tracks are filtered according to the detection confidence, with the integration of VOT in both forward and backward direction through the last few frames.

[54] proposed a unified framework with switcher-aware classification (SAC) in data association. A switcher is defined as the potential identity switch caused in the MOT context. The switcher-aware classifier, which is implemented using boosting decision trees, is employed to decide whether to use the features from the main target or the switcher to encode potential switcher information and improve robustness to handle the problem of identity switch. Both the long-term and short-term cues are gathered by the switcher-aware method and with potential switcher, scores for matching are generated. The Hungarian algorithm is used to match the tracklets with Kalman filter to smooth the final trajectories.

[55] proposed an instance-aware tracker that integrate VOT as a MOT solution via the encoding of awareness from both within and between target models with a dynamic model refreshing strategy. The awareness is implemented in both the target and global level, for each target, a KCF framework is applied, and features to distinguish target from background or other instances are fused. This method uses detection only for the purpose of model uniqueness verification to achieve spatial exclusiveness and model refreshing to update the model in response to the scale change of moving targets in detection. The difference between the target and the background as well as the difference among the different targets are modelled in the objective function of the proposed approach. The resultant events are "Tracked", "Occlusion", "Enter" and "Exit" for the targets observed, according to assumptions imposed on these events. For the case of occlusion, classification is performed and

followed by association via the Hungarian algorithm to obtain the final matching pair.

2.3.3 Using graphs that span many frames

[29] reviewed the classical Multiple Hypotheses Tracking (MHT) method and rehabilitate it with online discriminative appearance model, resulting in a method known as MHT-DAM in the MOT setting and achieved comparable performances with other SOTA methods at that time. The MHT method consists in building a graph with all possible object associations between frames given some constraints, and then finding paths in that graph that optimize a global score. Their version of MHT allows less restrictive assumptions on the motion of objects and thereby makes it less sensitive to the choice of parameters in the framework. With a tracking-by-detection approach, the MHT maintains multiple trees that represent all hypotheses from a single observation. They are kept active until the ambiguities of data association are resolved.

In order to handle complex interacting objects in MOT, [56] proposed Causal And-Or Graph (C-AOG) to utilize the causal relations between object's visibility fluent and its activities. In mathematics terms, a fluent [57] is defined as the time varying status of an object. In the C-AOG graph, there are four levels: 1) visibility fluents, 2) possible states and agent actions, 3) the Or nodes representing the alternative causes in visibility fluents and state levels, and 4) the And node representing the event that can encompass several atomic actions. A probabilistic model is applied to reason about the change from visible to non-visible or vice versa.

[36] proposed an end-to-end graph network that captures appearance and motion similarity separately. Unlike its static graph predecessors, the proposed method allows the updating of nodes, edges and global variables. A four-step graph network encompassing edge updating module, node updating module, global updating module and edge updating module was introduced in the proposed method. Hungarian algorithm is also used by [36] for data association with the dynamic graph. For handling missing detections, a VOT is used to track the missing objects in the current frame and perform association with recovered bounding boxes with high confidence score. A linear motion model is used to recover missing detections for a longer period of time.

Instead of using the tracking-by-detection paradigm, [43] explored the use of body joint detections in the MOT problem as it describes positions of person in a bottom-up way. The near online tracker was presented by solving MOT problem as a min-cost graph labeling problem with temporal sliding window.

[28] model the MOT problem as a minimum cost lifted multicut problem, where the lifted edges allow long-range information on the nodes to be joined or cut without modification on the set of feasible solutions. Long-term false joints are penalized by forcing valid paths in the feature solution. The proposed combination of both regular and lifted edges in the graph allows encoding of long-range person re-identification (ReID) information such that it forces valid paths along the local edges.

In the work by [41], hierarchical clustering of tracklets is achieved by formulating it as a minimum cost multicut graph problem. The tracklets are iteratively merged with repeated decomposition of a graph with their proposed Constrained Kernighan-Lin with Joins (CKLJ) algorithm that imposes constraints of joining the edges of the tracklets when dealing with high similarity edges.

In their proposed method, Structural Constraint Event Aggregation (SCEA), [58] introduced a cost function that consider global camera motion in the data association process with anchor assignment that makes sure that the centre location of an object coincides with the detection. Structural constraint and assignment events are also taken into consideration in the event aggregation to reduce assignment ambiguities from missing detections. All possible assignment between objects and detections are made based on the anchor assignment and structural constraints that are represented by location and velocity differences between objects. Instead of one-by-one matching, all the costs with the same assignment event but different anchor assignments are aggregated. A reduction approach that consists of gating and partitioning is later implemented to remove negligible assignments.

[31] proposed a non-uniform hypergraph learned automatically from a SVM that models different degrees of dependencies among tracklets, exploiting these cues in a computationally efficient way for MOT application.

2.3.4 Using deep learning

[59] jointly performed detection and tracking in a single neural network architecture with optimized parameters. An object detector, Single Shot MultiBox Detector (SSD) is adapted by serving the additional convolutional layer that outputs appearance features to a Recurrent Neural Network (RNN), combining them into tracks. The outputs of the detector is used as inputs to the RNN, with the use of an association metric (the Hungarian algorithm) integrated in the permutation layer. Track score based on track confidence, detection confidence and association confidence are thus computed.

[60] proposed the use of Recurrent Neural Network (RNN) in their fully end-to-end deep

learning approach to achieve online tracking of objects without prior knowledge, such as target dynamics or clutter distribution. The proposed method utilized the available public detections provided by the dataset. The RNN is used for temporal prediction and track management update and a long short-term memory (LSTM) network is used to solve the combinatorial problem of data association. [60] also uses additional variable in the loss function of the RNN to accommodate for the nature of the MOT task that has varying numbers of objects across frames. However, the accuracy of the tracking results is not as competitive as other existing methods.

Another end-to end solution, called Quadruplet Convolutional Neural Network (QUAD-CNN), was proposed by [61] for the MOT task by performing association of objects across frames with quadruplet losses, taking into consideration the appearance of objects and their temporal adjacencies. Additional constraints are imposed to make sure that the temporally adjacent detection are located more closely. In order to realize better localization, a multi-task loss is used to jointly learn object association and bounding box regression. Minimized label propagation is achieved in data association by utilizing learned distance metric from the paired detections.

For handling drift problems resulting from occlusion and interaction among targets, [23] proposed the Spatial Temporal Attention Mechanism (STAM) method that utilizes a spatial-temporal attention mechanism to favor possible pairings during data association. A motion model is also integrated in the method to capitalize on the motion information.

In the method by [62], a decision network made up of a reinforcement learning method is used to combine prediction and detection results to let each agent (object in the scene) maximize their shared utility (long term reward). The objects are either visible or invisible whereas the action sets are update, ignore, block and delete.

[63] proposed an early integration of the detection and tracking tasks for MOT via a two-stage detector that uses tracklet-based conditioning in both region proposal generation and classification. Both stages have Region Proposal Network (RPN) head added to perform tracklet-conditioning in their designed network. Association of tracklets is performed by a modified maximum bipartite graph matching.

In the work by [64], the Hungarian algorithm is used to assign the trajectories in a deep track association after the extraction of comprehensive appearance features and computation of their affinities. In their deep track association strategy, the track set is initialized with multiple trajectories with time stamps of the frame. The affinity matrices are computed efficiently using Deep Affinity Network (DAN) to look back into the existing tracks with each time stamp.

In order to capitalize on a visual attention mechanism that focus on most relevant regions for more discriminant features, [65] proposed Dual Matching Attention Networks (DMAN) that used both spatial and temporal attention mechanisms. The former mechanism generates dual attention maps that allow the network to focus on matching patterns of input image pair whereas the latter allocates different levels of attention to different samples in a tracklet in an adaptive manner to suppress noisy observations and improve data association results. For the temporal attention network, a Bi-directional Long Short-Term Memory (Bi-LSTM) is used to predict the attention value without being influenced by the noisy samples.

[66] proposed the Markov Decision Process (MDP) framework to model the lifetime of an object as a MDP, solving the MOT problem as an ensemble of multiple MDP. The similarity function of data association is equivalent to policy learning for the MDP. Taking into account all the possible transitions of states that represent the targets over time, [66] proposed different reward functions for policies in active state, tracked state as well as lost state. The MDP is viewed as an inverse reinforcement learning problem, where reward function is learned from the ground truth trajectories. The Hungarian algorithm is later used to perform the final data association based on the similarity scores.

[67] proposed an RNN structure that jointly reasons about multiple cues over a temporal window, where the motion and interaction models used LSTM for handling long term occlusions. The proposed RNN structure is comprised of three RNNs for appearance, motion and interaction, which are combined by a target RNN that outputs the similarity between a target and a detection. Inputs for the appearance RNN are the results from appearance feature extractor whereas the motion RNN takes velocity vectors as input. For the interaction RNN, occupancy grids centred on a specific target is used as the input in order to incorporate previous motion of the targets, as well as the behaviour of the neighbouring targets. The computed similarity score used in data association is performed by reasoning on sequence of observations. [67] jointly trained their target RNN in an end-to-end manner with their component of their feature RNN (appearance RNN, motion RNN and interaction RNN). The output vectors of the feature component are concatenated and act as the input to the target RNN that learns the long-term dependencies of cues for data association, where the Hungarian algorithm is then used to perform optimal assignment, using the MDP framework proposed by [66].

For solving MOT in crowded area with long-term occlusions, [68] proposed a three-step process for the tracklets which involves generation, cleaving and re-connection using a Siamese Bi-Gated Recurrent Unit (GRU) for tracklet-to-tracklet association. In the tracklet generation step, detection candidates and tracked candidates are matched in bipartite graph with

the Hungarian algorithm, followed by the tracklet cleaving step that used a bidirectional output GRU to obtain reliable tracklet and split tracklets that are unreliable. Both forward GRU and backward GRU share the weights and the resultant distance from the features are used for the cleaving and re-connection steps. With the imposed temporal and spatial constraints on the set of cleaved tracklets, the remaining tracklets are re-connected using IOU.

[69] proposed a MOT strategy that formulates a network flow problem as a function of pairwise association cost. The directed network flow graph is formulated with a cost that model the interplay between birth, existence, death and association of detections in the scene. The video is broken down into chunks and solved as different linear programming (LP) problems, resulting in different sets of trajectories that are associated using bipartite graph matching.

[70] proposed the use of Recurrent Autoregressive Network (RAN) to couple an internal memory and external memory for each object trajectory from the objects in the frame. It characterizes the appearances and motion dynamics of targets over time. Each RAN corresponds to an object of interest and data association is performed via bipartite matching by computing the likelihoods of object detection as presented by the distribution models from the RAN.

[24] proposed an end-to-end method called FAMNet (Feature extraction, Affinity estimation and Multi-dimensional assignment) that refined in a single deep network where all the layers are differentiable. Following the formulation of Rank-1 Tensor Approximation (RITA) framework proposed by [71], hypothesis trajectories generated from detections are passed into FAMNet to compute local assignments, where the affinity sub-network in the form of Siamese networks is used to extract the features of candidate patches. Two levels of affinity are computed for each hypothesis trajectory: pairwise affinity and long-term affinity of hypothesis trajectory. [24] adapted a different iteration scheme from RITA to fit the process into the deep neural network framework. The tracking process is performed by integrating detection and VOT, where an anchor candidate (candidate in the middle frame of the batch) that misses detection will connect with the virtual candidate given as prediction of VOT.

[72] proposed Tracktor, a MOT paradigm that originated from a detector, where the regression head of a detector is exploited to perform temporal realignment of bounding boxes of objects of interest. The proposed method was designed to be hassle free and free of specific training or complex optimization, hence it is targeted for simple scenarios instead of occluded or crowded scenes. By pooling features on the current frame using the bounding box of previous frame, the IOU is used to decide if the track would be deactivated. To account for a new

object, the initialization is done using the detection as new trajectory with the assumption that it is not an occluded object from the existing active trajectories.

[73] proposed a point-based framework called CenterTrack that jointly detects and tracks. An object of interest is represented as a single point in the bounding box. The end-to-end trainable and differentiable tracker works by offsetting the center of the object from its current frame to its previous frame. The association is performed via a greedy matching which is later performed based on the distance between the predicted offset and a detected center point in the previous frame.

In an effort to promote real-time MOT, [74] proposed a shared model that jointly learns target detection and appearance embedding simultaneously in a single shot deep network. With a Feature Pyramid Network (FPN) architecture, feature maps at three scales are obtained from a forward pass through the backbone network. Fusing of feature map is performed after with prediction heads. The learning of appearance embedding is performed with the use of a cross entropy loss. The Hungarian algorithm is used for the assignment of tracklets.

[75] proposed the TrackletNet Tracker that combines temporal and appearance information in a unified framework with an undirected graph model. Tracklets are represented as vertices in the graph and the edge between them represents the similarities between the connected tracklets, where the measure of similarities known as connectivity is used in the multi-scale tracker. Assuming that the size of bounding boxes in adjacent frames remains unchanged and with epipolar geometry constraints, the cost function is reformulated into a least square problem during tracklet grouping.

For association of tracklets, [76] computes the IOU of the tracklets and perform tracklet removal with a threshold. The notion of anchor tube used in this method is extended from the concept of anchor boxes in object detection [77] with the inclusion of temporal dimension in the application of MOT. As a filtering process, a Tube None Maximum Suppression (TNMS) is used to cluster the detected tubes of the same categories into multiple groups by the IOU. The final tubes for each group that has the maximum confidence of being positive are used to update the trajectory. The Hungarian algorithm is then used to update the track set based on the IOU matrix.

[78] proposed Chained-Tracker (CTracker) that paired attentive regression results as well as integrated object detection, feature extraction and data association into an end-to-end solution. A chain node is defined as two adjacent frames. Each frame is being used twice as part of different bounding box pairs known as chain node, resulting in a group of chain nodes being fed into the tracker. To generate long trajectories, node chaining is done sequentially over all adjacent nodes, where IOU between the boxes are computed and the Hungarian

algorithm is used to match the detections between different pairs of frames. The resultant box pair for each target from the paired boxes regression branch is achieved by simultaneous regression of two boxes with chained-anchors, which are densely arranged on spatial grid. In the tracker, a joint attention module was introduced with its predicted confidence maps, highlighting informative spatial regions with two other branches. One branch, known as object classification branch, works by predicting the confidence scores of the first box among the detected box pairs, where the scores guide the regression branch to the foreground regions. The other branch, known as ID verification branch, produces prediction that allows the regression branch to focus on regions corresponding to the same target. Due to the nature of node chaining, a memory sharing mechanism is employed to reuse the features to reduce computational complexity. Extracted features of the current frame are temporarily saved and reused until the processing of subsequent nodes.

[25] proposed RetinaTrack that originated from the single stage RetinaNet approach by [79] via some effective modifications on its post-FPN prediction subnetworks. For the purpose of tying or untying weights in a different manner than the vanilla architecture, the detection anchors are forced to split among the post-FPN prediction layers to access the intermediate level features. A third sequence of task-specific layers are applied to project instance-level features to a final track embedding space. The strategy of predicting instance-level features per-anchor is followed by a greedy bipartite matching of embedding vectors to associate the objects across frames using a track store that holds stateful track information.

Finally, [80] defined a fully differentiable framework based on Message Passing Networks (MPN) by exploiting the natural graph structure of the MOT problem to perform both feature learning, as well as final solution prediction. Direction prediction is learnt to find the edges in the graph that will be active for data association via classification. Time-aware update rules are proposed to aggregate the graph into two parts: one over the nodes in the past, and another one over the nodes in the future.

2.4 Object Description

Various features and embeddings (mappings in CNN that represent discrete variables as continuous vectors) have been proposed to provide the bases for the methods proposed in this work, starting from the conventional colour histogram, spatial and motion features to the more recent ReID features that are gaining popularity recently. These are mainly used together to give the best representation of targets to be tracked in MOT solutions.

2.4.1 Handcrafted features

[29] used pixel appearance features obtained from bounding boxes of a trained model of multiple linear regressors in their proposed MHT-DAM framework. The incorporation of appearance likelihood is reported to have significantly reduced the number of branches in their method, thereby achieving effective pruning of hypothesis to save memory.

[45] proposed the fusion of multiple features in a Dempster-Shafer framework in MOT. A sparse appearance model and a color model are used in this method. A score matrix to compare the similarity between two consecutive frames is computed and the accumulated score vector from each feature is used in a combined probability map with Dempster-Shafer rule of combination. The method uses detections obtained from an object detector based on classic machine learning techniques with HOG features and an SVM. The quality of the detections were not very good at that time, but detection of pedestrians (single object class) were sufficiently good for tracking.

The tracker by [47] uses appearance information that are computed as the average value of Kullback-Leibler distance of color histograms, in addition to motion coherence that is modeled by the smoothness of manifold fitted to the joint set between the track spatial history and new detected point.

In the tracker by [20], collection of keypoints of the extracted foreground regions were used to interpolate locations of targets. These keypoints, extracted from Binary Robust Invariant Scalable Keypoint (BRISK) [81] and described using Fast Retina Keypoint (FREAK) [82], are used with the extracted foreground blobs to better discriminate the objects in the scenes for handling occlusion, splitting, merging and fragmentation in a robust manner. A saved history of appearance information and positions for the tracks are also used to determine the transition of track states.

In the work of [22], complex features are avoided on purpose to minimize complexity in general and to allow fast and reliable implementation in real-time tracking. For handling short-term occlusion, IOU distance on the bounding box coordinates are used to correlate the occluder with the detection, whereas the covered object is unaffected and remained as an unmatched object.

2.4.2 Learned features

[48] extended the work of [22] by integrating an appearance descriptor from a pre-trained CNN and motion information to allow tracking during longer occlusion. In DEEPSORT, [48] integrated a learned association metric on a large scale people ReID dataset as a pretrained

step to further improve the tracking performance. The Mahalalobis distance computes the distance between predicted objects with a Kalman filter and newly detected objects by taking into consideration the state uncertainty via the standard deviation of the detection from the mean track location.

[46] proposed the Person of Interest (POI) tracker where distance between deep learning-based appearance features is used as the affinity score with a cosine distance used to measure it. In addition, motion and smoothness affinity are also used to obtain the final affinity between tracklets and detections.

In the work by [59], spatial distance, object appearance features provided by the deep network of the detector, detection score and track score are combined as the affinity metric.

Focusing on interaction of objects of interest with the surroundings objects, [56] introduced a fluent variable that denotes visibility status over time for objects of interest in their proposed method: visible, partially or fully occluded and contained. The fully occluded state and the contained states are distinguished by observing the motion independence, coupling actions and object fluent changes and visibility in the alternative view points. For the tracklet generation, objects' position and appearance from the average pooling of image descriptor over time of deep network is used.

[83] proposed spatial-temporal relation network (STRN) that uses representation of tracklet-pair similarity. Combination of cues includes deep appearance from CNN, location and topology over long period of time in the computation of similarity scores to capitalize on the information in both spatial and temporal domains. The spatial-temporal relation network is initially applied on each frame to strengthen appearance representations in the spatial domain, which are then aggregated in the temporal domain. The aggregated feature is then concatenated to represent tracklet-object pair and produce a similarity score.

[49] introduced a structural invariance constraint that jointly reasons about color histogram features and structure cues without having to manually adjust their respective parameters.

The work by [66] used appearance, location, size and history of the target as feature representation in their proposed framework. The appearance model is constructed by using an image patch of target in a video frame as a template. The similarity between a target and detection is encoded by the feature vector based on the status and the history of target from the reinforcement learning algorithm. Constrained optical flow information in a neighbourhood from each template is used as well.

In the work by [84], extracted appearance features from fully connected layer of CNN are the inputs that are multiplied with the LSTM memory matrix in their proposed Bilinear LSTM

architecture.

[21] proposed a deep appearance learning method that learns a discriminative appearance model of objects. During the learning process, pairs of the same objects and different objects are used as the input.

In the work of [85], the video is fed into a two-stream backbone structure to obtain 2D feature maps and a spatial-temporal feature map, which are then squashed and concatenated.

[64] proposed Deep Affinity Network (DAN) to jointly extract pre-detected features in multiple abstraction and pair the permutations of those features in any two frames to infer object affinities in an end-end manner. The two-stream convolutional network uses shared parameters to estimate the object affinities in later layers from initial layers without assuming the input frame pair to have similar appearance. In the feature extractor, layers of the network are empirically selected to reduce the number of feature maps and feature vectors from these layers are concatenated to represent an object. The affinity estimator works by mapping combinations of object features to a matrix that codes feature similarity, with gradual dimension reduction along the depth of the tensor. The method also takes into consideration the appearance and disappearance of multiple objects between video frames by enabling the softmax layer of the network to look forward and backward in time for the unidentified objects. The network loss is made up of four sub-losses: forward-direction loss, backward-direction loss, consistency loss and assemble loss.

TrackNet proposed by [85] incorporates a Tube Proposal Network (TPN) to predict the objectness of each candidate tube and their location parameters. Tube proposals are chosen due to their ease for obtaining all spatial-temporal locations in one shot as well as availability of global and local context in the spatial-temporal domain. A spatial transformer is used to transform and concatenate features from different viewing angles. Objectness score are predicted in the tube classification module, where anchor tubes (generated from duplication of the same bounding boxes across multiple frames) with high overlap are selected as positive proposals. The tubes are ranked according to the objectness score and go through the second stage of classification and regression. Position offsets for the tube are refined and tube pooling is performed via the union of all bounding boxes in a proposal tube.

In the tracker of [73], tracking-conditioned detection requires the current frame, previous frame and a heatmap rendered from tracked objects as input, outputting center detection heatmap of current frame, bounding box size map and an offset map for the association process. Due to the nature of its point-oriented representation, the detections are conveniently rendered in a class-agnostic single-channel heatmap in the training process.

In the method by [36], appearance features are obtained from the output of a CNN and motion features are obtained from the displacement between objects and detections. [68] used appearance features from CNN as well as motion cues from the dimension of their bounding boxes for matching the tracklets in the first stage of their proposed method, which are further processed in subsequent stages. Temporal and spatial cues are used in the cleaving and re-connection stages.

In [62], the agents (objects in the scene) are represented by locations, learned appearance features from a deep network and trajectories. Similarly, visual and spatio-temporal cues are used by [41], but extracted from tracklets instead of detections to resolve lengthy occlusions. The pairwise detection visual appearance is learned by a CNN and the spatio-temporal feature is learned by two bidirectional RNNs to compute the similarity of compared tracklets. While the visual appearance features are independent of the sequence of the detections, the spatial-temporal features take it into account as it is fed into the RNNs in both forward and reverse direction.

In addition to appearance and motion features, [30] proposed the use of Aggregated Local Flow Descriptor (ALFD) that encodes relative motion using long-term interest point trajectories for tracklet pairwise affinity measure.

[37] used position in terms of IOU, appearance and optical flow as affinity metrics in their proposed MOT approach. The appearance affinity is obtained from re-identification feature vectors. Inspired by [30], [37] developed an optical flow affinity that uses the Lucas-Kanade sparse optical motion algorithm [86] that tracks the points regardless of the detections. The keypoints for the optical flow are obtained from training a stacked hourglass network of vehicles keypoints from Pascal 3D dataset [87].

[43] integrated person's pose in their tracker, hence in measuring the joint-to-body spatial feature, affinities involving Barycentric distance, x-y-offset, angle in the reference box and distance in the reference box, Euclidean distance and scaled distance of the pose keypoints are applied. Taking into account the difficulty of missing joint detection, a post-processing step is performed when associating the detections in sliding window to remove redundant detections of the same person.

In order to deal with unexpected camera motion, [58] proposed a structural motion constraint between objects that is represented by location and velocity difference between objects. [58] used their proposed structural constraint to recover missing objects caused by camera motion and occlusion in the second step of their two-step approach.

In the work by [23], deep features are obtained from the pooling of ROI and they are weighted

by spatial attention. For weighing the features, learning of a visibility map of each target is performed to infer a spatial attention map. While the spatial attention mechanism is mainly for feature extraction, the proposed temporal attention mechanism strikes a good balance between historical and current visual cues of the targets involved.

Prior to obtaining appearance similarity, [65] exploited motion cues to select candidate detections. Detections surrounding the predicted locations from VOT are considered as candidate detections if they are not covered by tracked target. The appearance affinity is subsequently measured between the detections and the observations from the target trajectories. A spatial attention network with Siamese architecture is applied to the features to capture the high level-information from the top layer of the convolutional network.

[88] used both visual and temporal features in their work to account for the object-motion and ego-motion. In the proposed architecture called STED (Spatio-Temporal Encoder-Decoder), for the encoder component, a GRU is used to extract temporal features from the bounding boxes from past frames, whereas a CNN-based encoder extracts motion features from optical flow directly. The compact representation of the history of bounding boxes from the former is combined with the latter that contributes information from both object motion and ego-motion.

In the work by [28], deep features of paired identities are used as the input to their proposed tracking framework. Spatio-temporal relations, dense correspondence matchings and person re-identification matchings are the pairwise features used in the proposed method.

[70] used appearance and motion features as input to their tracking framework. The appearance features are extracted from the detector feature maps and the motion dynamics features are computed from the bounding boxes.

In the work by [24], the affinity sub-network fused discriminative higher order appearance and motion information into the affinity estimation. Similarly, in the work by [69], motion and appearance feature learnt from deep network in the detection are used in their proposed solution.

[89] proposed a light-weight sequential Generative Adversarial Network architecture that can be easily trained on limited data for predicting the trajectory of targets, taking into consideration the motion of pedestrian as well as the other pedestrians in the local neighbourhood, thereby modeling their interactions and contextual information. In the detection framework, the frames are passed through the encoder of the generative model. The LSTM layer maps the temporal relationships between the encoder embeddings of the appearance, and the decoder in turn maps them to a probability distribution that classified whether the

object is a pedestrian.

In the work by [72], a motion model and re-identification vectors are used as features. The proposed method was built on the assumption that the object changes only very slightly from one frame to another, hence camera motion compensation is used for sequences with moving cameras, and a constant velocity assumption is used on sequences with low frame rates to accurately produce bounding box positions on the targets. Stored deactivated trajectories and detections are compared in terms of distance and re-identified with a Siamese neural network. Similarly, the tracker by [25] included a ReID component and camera motion compensation to further capitalize on these information to improve the tracking performance.

For the affinity score in the tracker proposed by [21], a tracklet is represented by appearance, shape and motion models. The appearance model is obtained by the extraction of features of image pairs using forward propagation with L2 distance. This is achieved by minimizing the output feature distance of the same object pairs and maximizing the feature distance of the different pairs with online transfer learning to account for the variation of the data. The shape affinity is computed based on the height and width of the bounding boxes. The motion affinity is taken from the distance within the frame gap with an underlying assumption of Gaussian distribution.

In an effort to incorporate ReID features in the application of MOT with multiple cameras, [51] assembled a pipeline that uses a triplet loss function, focusing on the appearance features obtained from ResNet that underwent data augmentation, motion correlation and optimization.

The tracker of [54] incorporates both long-term and short-term cues. A short term cue is obtained from a subnet containing a VOT method, whereas a long-term cue is extracted from a ReID subnet. A quality-aware mechanism is employed to select the image sequence from the tracklet history with a quality filter in order to ensure quality and robustness. The long-term feature generation is achieved by feeding the selected images and the detection result to be matched into the ReID subnet.

In the work of [74], a tracklet is described with an appearance state and a motion state. The tracklet appearance is initialized from the appearance embedding from ReID network of the first observation. The motion state contains the bounding box centre position, bounding box height, aspect ratio and velocity. The cosine similarity is employed to compute the appearance affinity, whereas the Malalanobis distance is used to compute the motion affinity.

[55] used fused features that incorporates instance awareness that distinguishes between target and background as well as other instances. Feature map on the coordinates at the

centre of tracked target and their detection are extracted and stacked to be fed into a CNN-based classifier.

[53] used the IOU of the spatial overlap to describe the objects of interest in their tracking method. The work by [75] used appearance similarity and IOU of consecutive detections as embedded features in their proposed tracker.

[76] proposed a Deep Motion Modeling Network (DMM-Net) that estimate motion parameters of multiple objects to achieve joint detection and association in an end-to-end manner. The proposed network contains different sub-networks that predict object motion parameters, object classes and their visibility by exploiting the feature maps in the learned video sequence from a feature extractor based on the 3DResNet network by [90]. The network processes multiple frames simultaneously by introducing the notion of anchor tubes in the temporal dimension of MOT. The best matching track is searched and encoded into an anchor tube, it can be viewed as data preparation for DMM-Net. The motion model of the tracker is interpreted as a quadratic model in time and the motion loss function is the sum of smooth L1 losses between the ground truth encoded tracks and the predicted encoded tracks.

Finally, in the work by [80], appearance embedding from CNN and geometry embeddings from the relative position size and distance in time were concatenated and used for tracking.

2.5 Object Prediction

While some tracking methods already have some built-in prediction mechanism during the phase of data association, there are approaches that have explicit object prediction component in the tracking task to further fix the incorrect trajectories of the objects and to favor good potential matches. This is useful during occlusions where bounding boxes can be poorly positioned or absent. Several prediction methods can be used. We present some of them in the following.

In addition to data association, [22] used a Kalman filter to predict the bounding boxes of objects to further improve the accuracy of the tracked objects. A Kalman filter is state estimator that accounts for noise and missing observations.

The work of [50] utilized prediction at an early stage of their tracking method with the assumption that all targets move in an independent motion with constant velocity. In this case, the constant velocity model is used based on past observations. To handle the cases of possible target acceleration variations, noisy detections and camera motion that may generate erroneous predictions, the states of target over a large time interval are averaged.

In the work by [89], the motions of objects are predicted from a trajectory prediction frame-

work given by historical trajectories that passed through a LSTM encoder. For the motion prediction framework, a soft attention context vector is used to embed the trajectory information of the targets, and another hardwired attention context vector is used to embed the neighbouring trajectories.

In [89], the LSTM architecture of the encoder generates an output at each time step whereas the LSTM architecture of the decoder produces only one single output after considering a whole sequence. The short-term prediction is used for data association while the long-term prediction is used for updating the trajectory of the objects in the presence of occlusions and other image artefacts.

[62] proposed collaborative deep reinforcement learning (C-DRL) that simultaneously detects and predicts objects in a unified network using deep reinforcement learning. Each object is modeled as an agent and the prediction of their location is performed based on trajectory histories and appearance information using collaborative interaction between objects and the environment. A decision network is applied for the update, tracking and deletion of objects. For the prediction network, positive examples from all the training data are pooled together by merging the annotated ground truths and detections information, where IOU among them are considered valid.

[37] proposed the use of tubelet interpolation to handle cases of fragmentation in estimating the trajectories of objects. Using the optical flow information, the empty gaps of trajectories of lost objects are filled from the interpolation of bounding boxes, with the assumption that the movement of these objects follows a linear velocity model.

In the work by [88], the decoder component of their proposed method is used to generate predictions of future bounding boxes via a GRU, in which the concatenated features from the encoder are fed into.

In the RAN proposed by [70] that temporally model a generative framework for MOT, the external memory consists of input vectors in the previous time steps and the internal memory encodes information about the combination of template to predict the probability distribution of the next input.

The Bilinear LSTM by [84] acts as a building block of a predictor model for MOT application. The proposed algorithm uses Multiple Hypothesis Tracking (MHT) framework to predict whether the bounding boxes belong to each given track, using the learnt intuition from recursive least square for long term online predictions.

The tracker proposed by [74] used a Kalman filter to smooth the assigned trajectories by predicting the locations of previous tracklets in the current frame. Assignments that are

spatially too far away from the predicted locations will be rejected.

Finally, to enhance robustness against occlusions, CTracker by [78] retains terminated tracklets and their identities that are unmatched for a certain number of frames and a constant velocity model is used to predict the bounding boxes, the chain node process thereby includes these predictions for matching across the nodes.

2.6 Evaluation of MOT Performance

Performance of the tracking procedure are commonly evaluated using CLEAR MOT metrics [91], namely multiple object tracking accuracy (MOTA) and multiple object tracking precision (MOTP).

MOTA takes into consideration all configuration errors from all the frames by the tracker, including false positives, misses and mismatches. Mismatches are caused by the occurrences of mistaken swapping of objects' identities when they are close to one another.

$$MOTA = 1 - \frac{\sum_t (m_t + fp_t + mme_t)}{\sum_t g_t} \tag{2.1}$$

where m_t, fp_t and mme_t represents number of misses, number of false positives and number of mismatches respectively. g_t is the number of objects present at time t. Equation 2.2 to 2.4 are the ratio of the three errors respectively.

$$\overline{m} = \frac{\sum_t m_t}{\sum_t g_t} \tag{2.2}$$

$$\overline{fp} = \frac{\sum_t fp_t}{\sum_t g_t} \tag{2.3}$$

$$\overline{mme} = \frac{\sum_t mme_t}{\sum_t g_t} \tag{2.4}$$

MOTP evaluates the capability of tracker in estimating the precision of target without regards on its ability in recognizing object configurations and keeping consistent trajectories.

$$MOTP = 2 * \frac{\sum_{i,t} d_t^i}{\sum_t c_t} \tag{2.5}$$

where d_i^t is the distance in estimated position for matched object-hypothesis pairs and c_t is the number of matches made.

2.7 Summary

Much progress has been in development in recent years in the evolving field of MOT. In general, most MOT solutions follow an architecture that involves detection, appearance modeling and data association with prediction to obtain the final trajectories of objects of interest in the traffic scenes.

With the intensive progress and the surging rapid pace of deep learning techniques being introduced and applied in the application of MOT, we have observed that there is a recent trend that newly proposed methods combine the different modules in the framework concurrently, such as merging of detection and appearance modeling by an implicit appearance modeling like in Tracktor [72] and CenterTrack [73], achieving success of varying degrees.

For the extraction of objects, the trend of using deep learning-based detection approaches has dominated the more recent works as the problem of fragmentation can be avoided. Still, this choice prevents the tracking of unexpected objects. Therefore, ideally, both of this approaches would have to be combined.

Also, we have observed that more recent methods include deep features in addition to classical features, such as the combination of bounding box information with description of the appearance with deep features.

In terms of data association, the Hungarian algorithm remains a popular choice for online MOT application, whether it was used in the simplest conventional manner or in supplement to more complicated association schemes.

In order to further improve the performance of trackers, some works have additional predictive steps to fix the gaps in the final trajectories or perform some restoration on the evaluated track qualities. These component can greatly enhance the final outputs and critically removes potential problematic trajectories that can adversely affect the final results. In some works, prediction is also used in the data association stage to match the objects across frames.

CHAPTER 3 OVERVIEW OF THE METHOD

This chapter details the way the research problem of MOT in urban traffic is tackled and presents the development of our proposed solutions as they progressed and evolved over time, as documented by the three published papers.

We first started our work by devising a general MOT framework that underwent improvements over the course of this thesis. The classic MOT framework as illustrated in Figure 3.1 is used. The first paper presents the first version of our proposed MOT method, and the following papers iteratively improve its components.

Figure 3.1 The general MOT framework used in this thesis.

3.1 Article 1: Multiple Object Tracking in Urban Traffic Scenes with a Multiclass Object Detector

The initial idea of this paper stemmed from our earlier attempts at working on producing good inputs into a MOT in urban settings (see chapter 7). It was observed that most existing research works on road user tracking at that time were applying the unsupervised approach of foreground extraction to obtain the regions for the targets. There was a growing spike of interest with regards to the deep learning detectors that were gaining momentum with good reported results. We had an interest in evaluating their performance in the context of MOT and a motivation to integrate the results from the R-FCN detector [92] as the component for the extraction of targets, acting as input for our proposed tracker. Due to the limited size of data available for training, the detector was pretrained and refined on another dataset that is similarly on traffic scenes before being used to produce detection on our dataset of choice, Urban Tracker [20].

At the same time, given that our proposed urban traffic tracker deals with varying types of targets in the scene, we were inspired to capitalize on the class label information provided by the detector as the part of the feature combination for data association. The use of class labels in the proposed solution is very unique because most works focus on tracking a single object class (e.g. cars or pedestrians). In this project, several classes of objects are to be

tracked. An ablation study demonstrated the effectiveness of the class labels in improving the tracker performance. The data association component of the proposed tracker utilizes a Kalman filter for predicting the location of the targets in the subsequent frames whenever there is no matched detection with respect to the existing tracks, in similar fashion as the work by [22].

3.2 Article 2: Tracking in Urban Traffic Scenes from Background Subtraction and Object Detection

The results from Article 1 have inspired us to propose a strategy for fusing multiple sources of foreground extraction for our proposed tracker. Indeed, on the Urban tracker dataset, the detections of R-FCN were not very good. Therefore, in this paper, the combination of the detections from both background subtraction and a multiclass object detector is proposed. IOU is applied on the bounding boxes from these two sources to determine if they are indeed referring to the same object. At the same time, bounding boxes that are presumably missed out by one of the methods are added if they are matching in terms of IOU or colour similarity, which are the criteria for inclusion as a valid input for the tracker. With this, the proposed tracker is able to address the problems of fragmentation from the background subtraction approach as well as the elimination of the bounding boxes from the false detection originating from the supervised multiclass object detector.

For the feature combination, previously the Jaccard distance was used as the spatial distance. As a replacement, a different spatial feature formulation that averages the four corners of the bounding boxes is adopted, taking into consideration the dimension and scales that represent the objects as they move across frames. This allows better handling of bounding box size errors.

The overall quality of the tracks is also evaluated and applied according to a set of criteria to avoid the final tracking outputs being impacted by unreliable prediction.

3.3 Article 3: Supervised and Unsupervised Detections for Multiple Object Tracking in Traffic Scenes : A Comparative Study

As a continuation from the previous works, our proposed tracker is modified by including a new feature, a re-identification feature (ReID) into our feature combination since this type of feature was proven to be an effective indicator in many recent works. Effectively, our proposed tracker (now called MF-Tracker) utilized both classical features (spatial distance and colour histogram) as well as modern features from deep learning (detection labels and ReID). Instead

of R-FCN detector [92], the RetinaNet [79] detector, a more powerful and effective deep learning detection approach is employed on the existing dataset that was experimented before, that is the Urban Tracker dataset [20] as well as on a new dataset, the UA-Detrac dataset [3]. Given that the size of the dataset of UA-Detrac is sufficiently large for training and testing, more thorough experiments are performed to study the effects of supervised detection (deep learning detector) and unsupervised detection (background subtraction) for a documented comparison of their performances on MOT. Different combinations of baseline trackers with both types of detection as well as our own tracker are evaluated on both the Urban Tracker dataset and UA-Detrac dataset. This paper provides important insights on the advantages and drawbacks for deciding to use one of the two detectors.

CHAPTER 4 ARTICLE 1: MULTIPLE OBJECT TRACKING IN URBAN TRAFFIC SCENES WITH A MULTICLASS OBJECT DETECTOR

Hui-Lee Ooi, Guillaume-Alexandre Bilodeau, Nicolas Saunier and David-Alexandre Beaupré,
published at International Symposium on Visual Computing (ISVC), August 2018

Abstract

Multiple object tracking (MOT) in urban traffic aims to produce the trajectories of the different road users that move across the field of view with different directions and speeds and that can have varying appearances and sizes. Occlusions and interactions among the different objects are expected and common due to the nature of urban road traffic. In this work, a tracking framework employing classification label information from a deep learning detection approach is used for associating the different objects, in addition to object position and appearances. We want to investigate the performance of a modern multiclass object detector for the MOT task in traffic scenes. Results show that the object labels improve tracking performance, but that the output of object detectors are not always reliable.

4.1 Introduction

The objective of multiple object tracking (MOT) is extracting the trajectories of the different objects of interest in the scene (camera field of view). It is a common computer vision problem that is still open in complex applications. This paper deals with one of these complex applications, urban traffic, that involves different kinds of road users such as drivers of motorized and non-motorized vehicles, and pedestrians (see Figure 4.1). The various road users exhibit different properties of moving speeds and directions in the urban environment. Their size vary because of perspective. Besides, road users are frequently interacting and occluding each other, which makes it even more challenging.

In this work, we want to investigate the performance of a modern multiclass object detector [92] for the MOT task in traffic scenes. We are interested in testing MOT in urban traffic settings with road users of varying sizes using an object detector while most previous works in such applications employ background subtraction or optical flow to extract the objects of interest regardless of their size. Our contributions in this work is an assessment of a typical model object detector for tracking in urban traffic scenes, and the introduction of label in-

formation for describing the objects in the scenes. Due to the variability of objects found in urban scenes, the label information should be a useful indicator to distinguish and associate the objects of interests across frames, thereby producing a more accurate trajectory. In this paper, the improvements obtained thanks to classification labels are evaluated with respect to a baseline tracker that uses a Kalman filter, bounding box positions and color information.

The results show that using classification labels from a detector improves significantly tracking performances on an urban traffic dataset. Therefore, multiple object trackers should capitalize on this information when it is available. However, they also show that the outputs of a multiclass object detector are not always reliable and not always easy to interpret.

4.2 Related Works

MOT in urban traffic scenes was previously studied in [20], where the use of background subtraction is proposed for detecting the objects of interest followed by updating the object model with a state machine that uses feature points and spatial information. In fact, most previous work in MOT uses background subtraction or optical flow to detect the objects. The reason is that historically, methods based on pre-trained bounding box detectors are difficult to apply to road user tracking scenarios because it is difficult to design a detector that can detect and classify every possible type of road user from various viewpoints. However, recent progress in deep learning [92, 93] make this avenue now possible and worth investigating.

When using background subtraction, the detection results give blobs that can correspond to parts of objects, one object, or many objects grouped together. The task is then to distinguish between merging, fragmentation, and splitting of objects. This is the main drawback of this method, since under congested traffic conditions, road users may partially occlude each other and therefore be merged into a single blob. Examples of trackers based on background subtraction include the work of [94], [95], [96], [97], [98], and [20]. For data association, they typically use the overlap of foreground blobs between two frames or a graph-based model for data association using appearance information, such as textures, color or keypoints. These approaches track objects in a merge-split manner as objects are tracked as groups during occlusion. The Hungarian algorithm is a classical graph-based choice for solving object assignment problems. To compensate for the missing detections, the Kalman filter is a popular option for estimating the location of the object of interest. A basic implementation of multiple object tracking is proposed in [22] using this approach.

With optical flow, objects are detected by studying the motion of tracked points in a video. Feature points that are moving together belongs to the same object. Several methods ac-

Figure 4.1 A frame from the urban traffic dataset that shows several road users in an intersection.

complish this process using the Kanade-Lucas-Tomasi (KLT) tracker [99]. The following researchers have proposed such trackers, often called feature-based: [100], [101], [32] and [102]. For example, the algorithm proposed by Saunier et al. [32], named Traffic Intelligence, tracks road users at urban intersections by continuously detecting new features. The main issue is to select the right parameters to segment objects moving at similar speeds, while at the same time not oversegmenting smaller non-rigid objects such as pedestrians. Because objects are identified only by their motion, nearby road users moving at similar speed are often merged together. The exact bounding box occupied by the road user is unknown because it depends on the position of sparse feature points. Furthermore, when an object stops, its features flow becomes zero and feature trajectories are interrupted, which leads to fragmented object trajectories. Using a deep learning-based detector on road users is expected to provide objects that are less fragmented and that can be tracked whether they are moving or not.

4.3 Method

The proposed method consists of two main components: object detection and data association. It is illustrated in Algorithm 1. Object detection involves the extraction of objects of interest from the frames for further processing. Data association determines the tracking architecture to ensure the formation of the trajectories of each object in the scene. In order to match the objects correctly, an assignment cost based on a measure of similarity is computed for all the potential matches.

4.3.1 Object Detection

The road users from each frame are detected by using a deep-learning object detection model from the Region-based Fully Convolutional Network (RFCN) [92] framework due to its efficiency and accuracy. This detector was selected because it was the best performing approach on the MIO-TCD localization challenge [103]. The pre-trained model is further refined by using the MIO-TCD dataset [103] to provide the labels of the different road users found in traffic scenes, belonging to one of the eleven categories or labels: articulated truck, bicycle, bus, car, motorcycle, motorized vehicle, non-motorized vehicle, pedestrian, pickup truck, single unit truck and work van.

A non-maximal suppression (NMS) method [104, 105] is applied to reduce the redundant detections of the same road users in each frame.

4.3.2 Data Association

The object assignment or data association is essentially performed on a set of detected objects from the current frame and a list of actively tracked objects that are accumulated from previous frames.

For the matched pairings, the latest position of the corresponding object in the track list is updated from the detected object. In the case of new detection, a new object will be initialized and added to the track list. In the case of objects in the track list without a matched candidate from the detection list, i.e. a missing detection, a Kalman filter [106] is applied to predict its subsequent location in the scene and the track information is updated using the prediction.

For the matching of objects across frames, if the total cost of assigning object pairs is higher than a set threshold T_{match}, the paired object would be reassigned to unmatched detection and unmatched track respectively due to the high probability of them not being a good match.

Actively tracked objects that are not assigned a corresponding object from the new detections after $N_{timeout}$ frames are removed from the list, under the assumption that the object has left the scene or the object was an anomaly from the detection module.

Object Assignment Cost Once objects are detected, the subsequent step is to link the correct objects by using sufficient information about the objects to compute the cost of matching the objects. The Hungarian algorithm [107] is applied to match the list of active objects with the list of new detections in the current frame so that the matchings are exclusive

Algorithm 1 MOT algorithm

```
1: procedure MOT
2:     for i^th frame do
3:         Extract detections with multiclass object detector
4:         if i == 1 then
5:             Assign all detections as tracks
6:         else
7:             for each detection do
8:                 Compute cost of detection with respect to each track
9:             Run the Hungarian algorithm for assigning pairing of detection and track
10:            for each matched detection do
11:                if Cost > T_match then
12:                    Reassign as unmatched detection and unmatched track
13:                else
14:                    Update the track information from the detection
15:            for each unmatched detection do
16:                Initialize as new track
17:            for each unmatched track do
18:                if N > N_timeout then
19:                    Remove track
20:                else
21:                    Update track information using prediction from Kalman filter
```

and unique. The bipartite matching indicates that each active object can only be paired with one other candidate object (the detection) from the current frame. The algorithm can make use of different costs of assignment, with higher costs given to objects that are likely to be different road users.

Label Cost In order to describe the properties of the detected objects, the labels and corresponding confidence score from the detections are taken into account. Setting the range of scores between 0 and 1, object pairs across frames that are more similar will be given a lower cost. Using the classification labels, object pairs with different labels are less likely to be the correct matchings, therefore they will be given cost of 1. Meanwhile, when the pairing labels are the same, the average of the confidence score of each detection are being taken as the label cost. The label cost is defined as

$$C_{label} = \begin{cases} 1 - 0.5 \times (\text{Conf}_i + \text{Conf}_j) & \text{if } L_i = L_j \\ 1 & \text{if } L_i \neq L_j \end{cases} \tag{4.1}$$

where L_n denotes the label of detection n and $Conf_n$ denotes the confidence of the corresponding label of the n^{th} detection.

Jaccard Distance-based Position Cost The bounding box coordinates of the detected objects are a useful indicator for matching the objects across frames as well. To judge the similarity of two bounding boxes in terms of proximity and size, the Jaccard distance is computed from the coordinates of the paired object, where the ratio of intersection over union of the bounding boxes is computed. This is calculated using

$$C_{position} = 1 - \frac{|Box_i \cap Box_j|}{|Box_i \cup Box_j|} \tag{4.2}$$

where Box_n denotes the set of pixels of the bounding box of the detected object n.

Color Cost The visual appearance of the objects is characterized by their color histograms that are used to compute the color cost. In this work, the Bhattacharyya distance is applied to compute the distance of the color histogram of detections across frames with

$$C_{color} = \sqrt{1 - \frac{1}{\sqrt{\bar{H}_i \bar{H}_j N^2}} \sum \sqrt{H_i H_j}} \tag{4.3}$$

where H_i denotes the color histogram of detection i, H_j denotes the color histogram of detection j and N is the total number of histogram bins.

4.4 Results and Discussion

To test the RFCN multiclass object detector in MOT and to assess the usefulness of the classification labels, we used the Urban Tracker dataset [20] since it contains a variety of road users in an urban environment. Fig 4.2 shows some sample frames from the Urban Tracker dataset with RFCN detections. The MOT performance is evaluated by using the CLEAR MOT metrics [91]:

- multiple object tracking accuracy (MOTA) that evaluates the quality of the tracking, if all road users are correctly detected in the frames they are visible and if there are no false alarms;
- and multiple object tracking precision (MOTP) that evaluates the quality of the localization of the matches.

To test the contribution of using labels in MOT, the proposed baseline method is applied with and without object classification labels in the cost computation for data association. The following parameters are used in the experiments: T_{match} is set at 1.5 and the value of $N_{timeout}$ is set at 5.

Table 4.1 summarizes the results obtained with the baseline tracker. First of all, we were not able to obtain interesting results on the René-Lévesque video. From the evaluation, it is observed that the size of the objects greatly influences the performance of the proposed method because of the shortcomings of RFCN. When the size of the road users is too small, there are not enough details for the detector to distinguish the different types of objects reliably. Mis-detections are common in such cases, as observed in video René-Lévesque, for example in Figure 4.3. Since the frames are captured at a higher altitude than the other urban scenes, the object detector has difficulties in detecting and classifying the objects clearly due to the lack of details. On the other hand, larger objects such as buildings have the tendency of being detected as they share similarities with the features learned by the detector.

Secondly, it can be noticed from Table 4.1 that the MOTA results are negative and disappointing. This comes from the difficulty of interpreting the detections of RFCN. The same object is sometimes detected as several instances from the object detection module, as shown in Figure 4.4. This often causes confusion and unnecessarily increases the number of detected

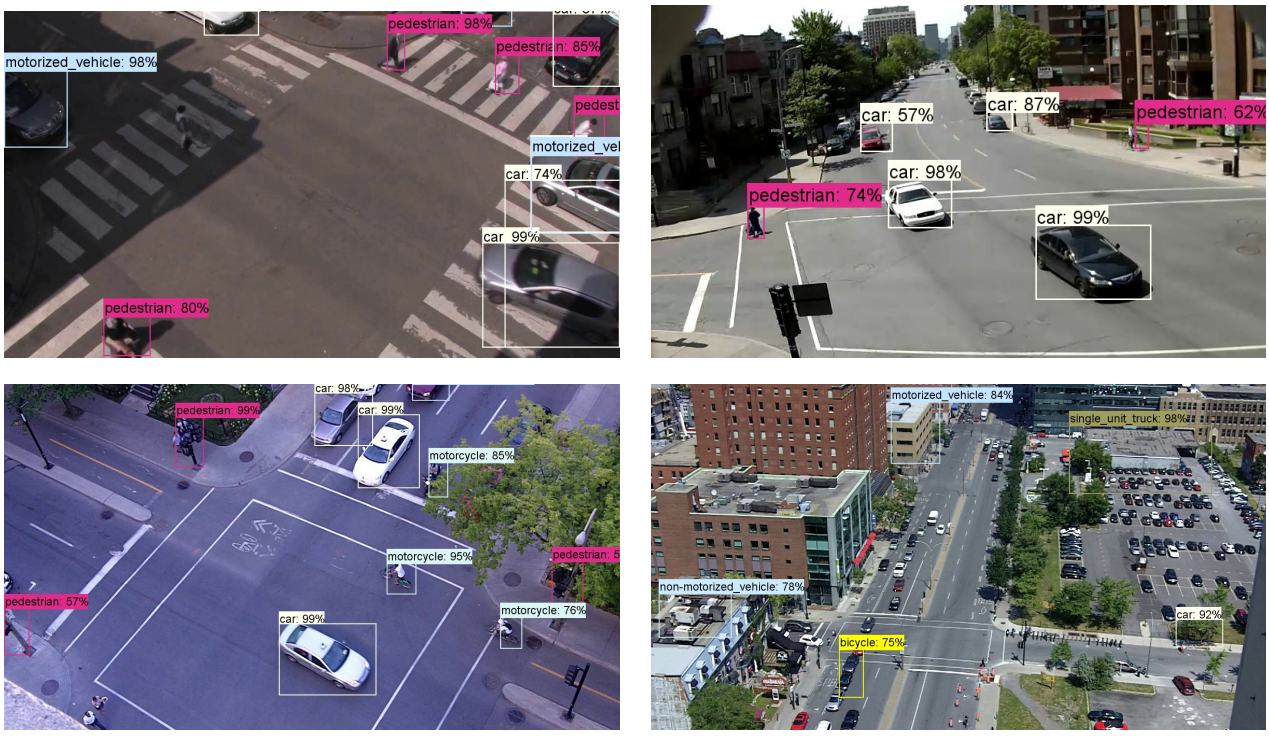


Figure 4.2 Samples frames with detections from the Urban Tracker dataset

Figure 4.3 Typical detections obtained from the René-Lévesque video.

objects and degrades the reported tracking performance. When there are no consecutive redundant detections, these redundant instances of the same object will usually be removed after a few frames since the object assignments are exclusive.

Furthermore, contrarily to background subtraction or optical flow-based methods, RFCN gives detection outputs also for cars that are parked or for a car on a advertising billboard.

Table 4.1 Comparison of MOTA and MOTP scores for three videos of the Urban Tracker dataset with the inclusion and exclusion of label cost in the data association (the best results are in boldface).

	No. objects	MOTP		MOTA	
		with labels	without labels	with labels	without labels
Rouen	16	0.6870	**0.6893**	**-0.1877**	-0.4176
Sherbrooke	20	**0.7488**	0.7324	**0.0266**	-0.0023
St-Marc	28	**0.7234**	0.7136	-0.3657	**-0.2749**

Therefore, the data association process is distracted by many irrelevant objects. In such cases, standard NMS is not very useful in a traffic scene. Although NMS is used, it is insufficient to eliminate all the redundancies.

Since the proposed method is intrinsically dependent on the results from the detection module, the mis-detections propagate and deteriorate the overall MOT performance. In this case, the existence of redundant tracks severely affects the MOTA score such that it falls into the negative range, as shown in Table 4.1. The MOTA takes into account the number of misses, false positives and mismatches from the produced trajectories.

However, it can be noted that MOT with inclusion of classification label generally gives higher MOTA. Among the different classes of labels from the detection module, the non-motorized vehicle label is currently excluded in the tracking framework since the occurrence of non-motorized vehicles is very rare in this experiment while parts of the background are sometimes mistakenly identified as objects from this class. MOTP is sometimes slightly better without labels as there are cases where tracking of an object fails in successive frames due to the switch of labels from the detection results. This is because with the labels, some matches are penalized and rejected because they are higher than the cost threshold. Therefore, the total number of matches is different, leading to slightly different values for MOTP. This occurrence is common among classes that share similarities such as pedestrians, bicycles and motorcycles, resulting in redundant tracks or fragmented tracks for the same object and thus lowering the overall tracking performance.

4.5 Conclusion

In this paper, the use of a modern multiclass object detector was investigated for the MOT task in traffic scenes. It was integrated in a baseline multiple object tracker. Results show that classification labels can be beneficial in MOT. However, the outputs of the multiclass

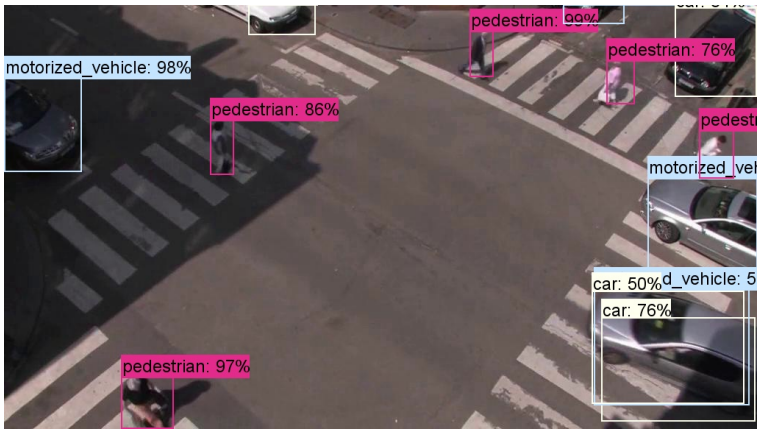


Figure 4.4 An example of the redundant detection output for the same object.

object detector are hardly usable because they include a large number of false detections, or detections of objects that are not of interest in the current application (e.g. parked cars). Small objects are also difficult to detect. As a result, to use such a detector, its output needs to be combined with another detector that can focus more precisely on objects of interest such as background subtraction or optical flow.

4.6 Acknowledgement

This research is partly funded by Fonds de Recherche du Quebec -Nature et Technologies(FRQ-NT) with team grant No. 2016-PR-189250 and Polytechnique Montréal PhD Fellowship. We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan X GPU used for this work.

CHAPTER 5 ARTICLE 2: TRACKING IN URBAN TRAFFIC SCENES FROM BACKGROUND SUBTRACTION AND OBJECT DETECTION

Hui-Lee Ooi, Guillaume-Alexandre Bilodeau and Nicolas Saunier,
published at International Conference on Image Analysis and Recognition (ICIAR), May 2019

Abstract

In this paper, we propose to combine detections from background subtraction and from a multiclass object detector for multiple object tracking (MOT) in urban traffic scenes. These objects are associated across frames using spatial, colour and class label information, and trajectory prediction is evaluated to yield the final MOT outputs. The proposed method was tested on the Urban tracker dataset and shows competitive performances compared to state-of-the-art approaches. Results show that the integration of different detection inputs remains a challenging task that greatly affects the MOT performance.

5.1 Introduction

The task of multiple object tracking (MOT) is to produce a set of trajectories that represent the actual real-life movements of the objects of interest across frames. In the context of urban scenes such as traffic intersection, MOT is performed for the road users (vehicles, pedestrians, cyclists, motorcyclists, etc.) as objects of interest for the purpose of traffic control and management to improve traffic while mitigating the adverse impacts. Due to the nature of such settings, interactions among the objects are expected and frequent, thus leading to object occlusions. Compared to conventional traffic scenes where the speeds of the road users are usually more consistent and directions homogeneous, MOT in urban traffic scenes remains a difficult and challenging task as it deals with objects interacting in different directions and speeds. Furthermore, because of the typical camera setups used, object scales varies significantly, which can make them difficult to detect.

The advances and reported good results in recent years of multiclass object detection algorithms with deep learning [108] have prompted us to integrate them into the tracking process. In addition, the class label information can provide a useful description of objects to help with their association across frames in the tracking steps. However, the recent work of Ooi et al. (Chapter 4) has shown that tracking with a multiclass object detector (MOD) is very challenging since detections are often incorrect or missing. Since the incorrect or missing

detection of an object at that stage can propagate and leave a huge impact on the final tracking results, we seek to improve the detection inputs in order to achieve better MOT results. Therefore, we extend the work of Ooi et al. (Chapter 4) by using as inputs, detections from both a MOD and a background subtraction algorithm. To handle the problem of occlusion, a Kalman filter is used for prediction when an object of interest is not seen at the detection stage. This helps in keeping track of object of interest that might have been hidden by other objects at certain time steps during the lifespan of the trajectory.

In this paper, we introduce a MOT solution for urban traffic scenes with fused inputs from the integration of background subtraction inputs [35] with detections from a pre-trained MOD. Our two main contributions are: 1) a novel method to fuse detections from two sources that may contradict each other and 2) an object descriptor based on object class labels and their learned detection confidence.

5.2 Related Works

MOT usually comprises several steps: 1) object detection, 2) appearance modeling, and 3) data association. A large part of the past literature on MOT emphasized the challenge of data association [22] and its effect on MOT performance. Researchers proposed sophisticated data association strategies that often extend the Hungarian algorithm. For example, the Joint probabilistic data association filter (JPDAF) tracks objects based on the most likely outcome for each trajectory by considering every detection available, as well as missing or spurious detections [109, 110]. Another example is the minimum-cost flow algorithm that formulates the data association problem as finding the shortest path from the apparition of the object to its last appearance in the scene [111].

On the other hand, object detection is necessary before data association, as poor detections will severely deteriorate the tracking performance. Hence, some previous MOT solutions have proposed combining detection methods to allow better object inputs for improved tracking in the end. The main drawback of using inputs from background subtraction is the difficulty of distinguishing the merging, fragmentation and splitting of objects. In cases where multiple road users are in close proximity, partial occlusion will cause the incorrect merging of these road users. IMOT (Improved Multiple Object Tracking) was introduced by [35] as an improved version of background subtraction using edge processing and optical flow, converting blobs of objects into compact bounding boxes that outline individual objects if there is evidence based on motion that two or more objects were grouped together.

More generally, the MOT problem in urban scenes was tackled several times in the past. A

combination of background subtraction and feature points were proposed in Urban tracker [20]. Based on detections from background subtraction, objects are described by several keypoints which provide robustness to partial occlusion as a subset of keypoints can be matched if they are not all hidden. MKCF [34] was proposed as a solution for MOT, combining the background subtraction with multiple individual KCF (Kernelized Correlation Filters) single object trackers [112]. This method capitalizes on the robustness of newer visual object tracker. It shows good performances even if it uses rudimentary data association. [32] used optical flow to detect the motion of objects of interest and the classic Kanade-Lucas-Tomasi (KLT) framework [99] to match road users from frame to frame. Recently, Ooi et al. (Chapter 4) used a MOD for road user tracking. However, the tracking performance was severely impacted by the inadequate and inconsistent detections across frames.

5.3 Methods

Three steps are involved in the proposed MOT strategy: (i) Fusion of objects from detection methods, (ii) Object description, and (iii) The association of objects across frames. Our proposed method is illustrated in Figure 5.1. It starts by fusing the input from a multiclass object detector (MOD) and from the improved background subtraction method IMOT. The resulting object detections are then tracked. Objects are described using colour, position and the class labels coming from the MOD. Then, data association is performed.

5.3.1 Object fusion

In our proposed method, we integrate the bounding boxes from both IMOT and MOD into our tracking framework. The MOD objects are the result of the application of a pre-trained deep learning detection network, in our case RFCN [108], that was fine-tuned on the MIO-TCD dataset [103] containing varied road users such as cars, buses, bicycles, pedestrians, pickup trucks, etc. IMOT objects are the results of a post-processing over a background subtraction method in order to separate erroneously merged road users using edges and optical flow [35].

The objects from the two sources are matched and filtered before starting the tracking process. Due to the nature of IMOT objects, there could be some small bounding boxes that are not relevant as a result of shaking cameras and moving background elements. On the other hand, MOD objects will include long-term stationary road users that are beyond the scope of interest of our applications and there are occasions where objects of interest are missed out (Chapter 4). We hypothesize that the merged inputs can be fed into the tracker to

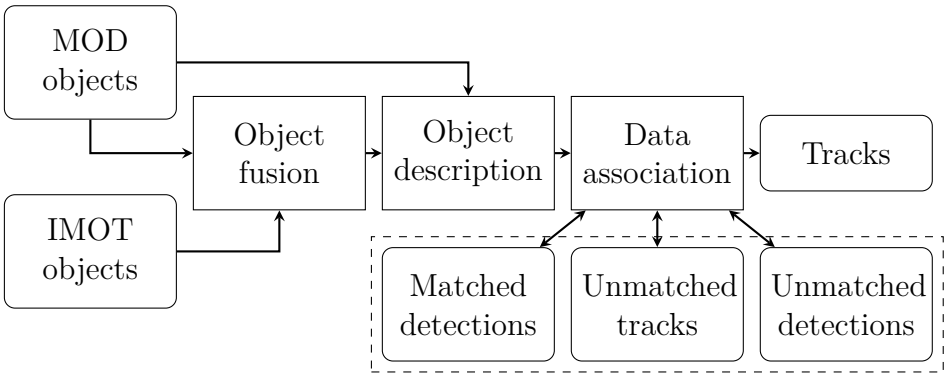


Figure 5.1 Overview of our tracking framework. Object detections from two methods are first fused. They are described and associated across frames using sets of matched and unmatched tracks and detections. Based on these, the final tracks are outputted.

give more satisfactory tracking results. However, results are often contradictory, sometimes IMOT gives better results while sometimes, it is MOD. One cannot simply merge the two sets of detections.

The following fusion strategy is proposed. We assume that all IMOT objects are relevant to the tracking framework as stray small IMOT objects that are not representative of objects of interest are filtered out according to size prior to the matching. IMOT objects were shown to be more reliable. They also had better performance in detecting the objects in the scene. MOD objects are used to provide class labels and to merge fragmented IMOT objects.

For matching the input objects of both sources, we compared their similarities in terms of bounding box (BB) overlaps and colour histogram. The BB overlap S_o is given by

$$S_o = \frac{A_i \cap B_j}{A_i \cup B_j}, \tag{5.1}$$

where A_i denotes the i^{th} BB from IMOT output whereas B_j denotes the j^{th} BB from MOD output. We also calculate the colour similarity between IMOT objects and between IMOT and MOD objects. The colour similarity S_c is given by

$$S_c = \sqrt{1 - \frac{1}{\sqrt{\bar{G}\bar{H}N^2}} \sum_{i=1}^{N} \sqrt{G_i H_i}}, \tag{5.2}$$

where G denotes the colour histogram of a first BB and H denotes the colour histogram of a second BB. N is the total number of histogram bins. $\bar{G}$ and $\bar{H}$ are the mean of the N bins.

Pairings between IMOT objects and MOD objects are performed based on the overlap of

the BBs with Equation 5.1 and a threshold T_o. IMOT objects that are matched with MOD objects will benefit from the class label information of MOD objects for data association in the tracking phase. On the contrary, IMOT objects that do not matched with MOD objects will be fed into the tracker with a dummy class label. There are cases where the matching is not one-to-one. For instance, several IMOT objects could be matched with the same MOD object and vice versa, though the latter is a rare occurrence since IMOT objects are usually smaller in size and more compact. MOD objects, on the other hand, are larger and often encompassed several objects at the same time. Hence, the merging of input objects is performed only on the IMOT objects and only if the colour of the objects to merge are similar enough based on Equation 5.2.

Algorithm 2 describes the process for obtaining the final fused inputs for the MOT task. If multiple IMOT objects are matched to a particular MOD object, the colour histogram similarities of the multiple IMOT objects will be compared among themselves using the Bhattacharyya distance (Equation 5.2) and a threshold T_c. If the similarity is high, then these objects are thought to be fragmented parts of an object and hence the BB from the MOD object would be taken as the input for the tracker. If the similarity is low, these objects will be considered individually in the subsequent steps. Figure 5.2 illustrates how fragmented parts of an object are fused together to recover the whole object after taking into consideration the output of the MOD.

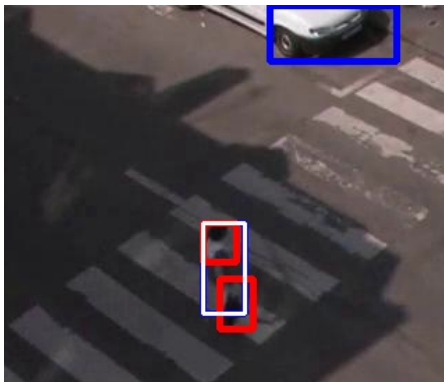


Figure 5.2 Example of the merging of objects. Blue BBs: MOD objects, red BBs: IMOT objects, white BB: resulting fusion of the two inputs into the whole object (pedestrian).

Colour similarity is used for merging IMOT because there are cases where the BB from the MOD object contains more than one actual object that should not be merged. Hence, care must taken to handle the different cases. To avoid excessive merging of IMOT objects that overlap with often large MOD objects, merging IMOT based on pairings between objects from the two approaches will only be evaluated if there is significant overlap (larger than T_m).

Finally, when a single IMOT object is matched with multiple MOD objects, the similarity in terms of BB overlap and colour histogram will be used to determine the final label from MOD that will be used with the IMOT object.

Algorithm 2 IMOT and MOD object fusion

```
1: procedure IMOT-MOD Pairing
2:     for each IMOT objects do
3:         for each MOD objects do
4:             Compute overlap of BBs with Eq. 5.1
5:             if S_o >= T_o then
6:                 Assign as pairs and update pairing matrix
7: procedure Merging multiple IMOT into single detection object
8:     for each MOD objects do
9:         if Pair with more than one IMOT then
10:            for each IMOT objects that are paired do
11:                Compute colour similarity with Eq. 5.2
12:                if S_o >= T_m and S_c <= T_c then
13:                    Use MOD object as tracker input, discard the IMOT object
14:                else Keep IMOT object
15: procedure Update IMOT object with label from paired MOD object
16:    for each Remaining IMOT objects do
17:        if No pairing found then
18:            Use the IMOT as tracker input with dummy label
19:        else if One-to-one pairing found then
20:            Use the IMOT as tracker input with label from paired MOD
21:        else
22:            for each Paired MOD objects do
23:                Compute S_c and S_o of IMOT with each MOD object
24:                Use IMOT as input with label of MOD object with largest similarity
```

5.3.2 Data association costs

The cost of assigning pairings among the objects across frames is calculated by using the Hungarian algorithm [107]. The cost of matching a detected object and a tracked object is in the range of 0 and 1. The lower the cost, the more likely the two objects are referring to the same object.

For matching the objects across frames, the spatial cost C_d is measured by the spatial distance

between BBs of the compared objects using

$$C_d = 1 - max(0, \frac{T_d - \bar{SD}}{T_d}) \tag{5.3}$$

$$\bar{SD} = \frac{1}{4}(|x_{D,min} - x_{T,min}| + |y_{D,min} - y_{T,min}| + |x_{D,max} - x_{T,max}| + |y_{D,max} - y_{T,max}|), \tag{5.4}$$

where x_{min} and y_{min} denotes the minimum x and y coordinates, whereas x_{max} and y_{max} denotes the maximum x and y coordinate of an object. T indicates an object that is currently tracked and D indicates a detected object in a frame. $\bar{SD}$ is the mean spatial distance of the x coordinates and y coordinates of the four corners of the BBs of the compared objects whereas a fixed parameter T_d is used to penalize objects that are too far and to normalize C_s.

For describing objects in terms of appearance, colour cost C_c is computed based on the Bhattacharyya distance on colour histogram as in Equation 5.2, where G denotes the colour histogram of a detection and H denotes the colour histogram of a currently tracked object. N is the total number of histogram bins (we used 256).

Finally, the class labels are also considered in the matching cost. Detection confidence is used in our formulation. C_l is given by

$$C_l = \begin{cases} 1 - 0.5 \times (W_i + W_j) & \text{if } L_i = L_j \\ 1 & \text{if } L_i \neq L_j, \end{cases} \tag{5.5}$$

where L_i denotes the class label of object i and W_i its confidence value (between 0 and 1). As we will see in the results, using the confidence value from the MOD, and not just the class label for the cost is a beneficial strategy since confidence values tend to be similar in consecutive frames for a given object.

The final association cost is a combination of C_d, C_c, C_l, and is given by

$$C_{final} = \alpha C_d + \beta C_c + \gamma C_l, \tag{5.6}$$

where α, β, γ denotes the weights for the corresponding cost.

5.3.3 Overall Tracking Framework

In the tracking phase, each input object that appeared at the start of the video will be included into a set that contains all the active objects, thereafter denoted as the tracked objects. New input objects in the subsequent frames are denoted as detected objects and are matched accordingly to the tracked objects. We enforce one-to-one matching using the Hungarian algorithm [107], since it is expected that there exists only one true object at the next frame that corresponds to a currently tracked object. In addition, because of the non-ideal cases caused by occlusions or objects missing from the inputs that are common in urban scenes, some predictions are used to compensate the shortcomings of the inputs.

Hence, for each processed frame, sets of matched detections, unmatched detections and unmatched tracks are obtained. Matched detections are essentially the successful pairings of detected objects and tracked objects. Unmatched detections refers to detected objects without pairing with the existing set of tracked objects. This can be due to the entrance of a new object into the scene or as a result of spurious detections from the inputs. Unmatched tracks are when there is no corresponding pairing found in the set of detected objects. This is usually due to occlusion or being missed by IMOT, but also by objects that have left the scene.

For each active tracked objects, a Kalman filter is used to get a prediction of its expected location in the subsequent frame based on its history. If the tracked objects are matched with the detected objects, the prediction result will be discarded and the tracked object will be updated with information from the latest matched detected object. In the case where a tracked object is unable to find a matching counterpart in the set of detected objects, the prediction result may be used instead if it is deemed good. For each step of a track, the state or quality of the tracking is defined as "D" (Detection), "GP"(Good Prediction), "BP" (Bad Prediction) or "UP"(Uncertain Prediction). Overlap between the prediction result and the previous position in the trajectory (history) is used to evaluate the quality of a prediction. If the previous time stamp is marked as "D" (indicating it is from a matched pairing) or "GP" (indicating it as a reasonably good prediction), there is a good chance that the trajectory history is reliable. Hence, if the overlap is high (larger than T_p) between the prediction at the current step and the previous history step, the prediction result will be used and the state will be marked as "GP". If the overlap is not good, the state will be marked as "BP" and the prediction result will not be used. Instead the previous result in the tracking history will be used. For cases of unmatched tracked objects with a history that is not marked "D" or "GP", the state will be marked at "UP" since there is no known reliable history that can be used to verify the current prediction. Algorithm 3 summarizes the inspection of the tracking

prediction quality.

Algorithm 3 Checking prediction quality for unmatched tracks

```
1: procedure PREDICTION QUALITY
2:     for each Unmatched track do
3:         if Previous time step is "D" or "GP" then
4:             if Overlap of prediction with previous time step < T_p then
5:                 Use BB output from previous time step and mark as "BP"
6:             else
7:                 Use prediction and mark as "GP"
8:         else
9:             Use prediction but mark as "UP"
```

At the end of the tracking process, trajectories with significant amount of "BP" and "UP" will be removed eventually since these final trajectories are likely to contain incorrect prediction that does not reflect the actual movement of the objects of interest.

For active track management, when a tracked object is unable to find a matching detection object for T_n frames, the object is assumed to have left the scene. The track will therefore be terminated along with its last T_n steps of tracking results removed since they are most likely not valid.

5.4 Experiments

The proposed method was tested on the Urban tracker dataset [20] and compared with several state-of-the-art methods. We also performed an ablation study on the data association cost components. The dataset includes four videos: Rouen, Sherbrooke, St-Marc and Rene-Levesque. We chose this dataset because it includes a large variety of object classes and background subtraction is applicable.

The tracking performance is evaluated by using the CLEAR MOT metrics [113] that are comprised of MOTA (Multiple Object Tracking Accuracy) and MOTP (Multiple Object Tracking Precision). MOTA evaluates the tracking performance by taking into consideration the number of objects that are mismatched, the false positives (FP) and false negatives (FN). MOTP evaluates the quality of the localization of the matches by checking the similarity of true positives (TP) with the corresponding targets in ground truths.

We also report the following information. Ground truth (GT) is the number of actual object instants in the whole video. Misses are missing GT object instances in tracks. FP are spurious object detections that are not in the ground truth. Mismatches are the number of tracks

that suffer from object identity switches. The identification of correct tracks, misses and FP are based on the overlap of bounding boxes from our tracking output with respect to the ones of the ground truth. We used a threshold of 0.3 for the overlap in tracking performance evaluation as proposed by citebeaupre2018improving. In our experiments, $T_o = 0.05$, $T_m = 0.5$, $T_c = 0.5$, $T_d = 0.5$ and $T_p = 0.01$.

5.4.1 Ablation study

We start our evaluation of our method with an ablation study on the data association cost. The individual effects of the cost components are compared in Table 5.1. Generally it is observed that the spatial cost has the smallest number of mismatches and FP for all the evaluated videos. Since the spatial cost is based on the proximity of BBs, it is an essential component that describes the similarity of objects to determine across frames. In the results for St-Marc and Rene-Levesque, it has the highest number of correct tracks compared to the other association costs.

Colour cost gives slightly inferior tracking performance, having more FP and mismatches with slightly fewer correct tracks compared to the spatial cost. This could be due to presence of multiple objects that share similar colour properties and the fact that proximity is ignored. In addition, since BBs contain a certain portion of background as well (depending how well the object is enclosed in the BB), this might not be the best cost component. However, it can disambiguate the association of nearby objects with different colours.

Lastly, the class label cost gives the lowest performing tracking results due to reasons similar to the colour cost. There could be several objects that share the same class label in the same frame. With only the class label information it is often insufficient to do the right pairings. Also, some IMOT objects are fed into the tracker with dummy class labels since they are not paired with MOD objects in the object fusion stage. Nevertheless, the performance with this feature is better than expected thanks to the similarity of confidence values for the same object between frames. Since, the confidence value is used in Equation 5.5, objects are both discriminated by their class and the confidence value.

5.4.2 Comparison with state-of-the-art methods

The performance of the proposed method is compared with previous state-of-the-art work, IMOT [35], Urban tracker [20], MKCF [34] and Ooi et al. (Chapter 4) that were evaluated on the Urban tracker dataset. For the data association cost, the weights of spatial, colour and label costs are 0.6, 0.3 and 0.1 respectively for α, β and γ. As shown in Table 5.2, the

Table 5.1 Comparison of individual association cost components for the four videos of the Urban tracker dataset. **Boldface** indicates best result.

	Cost	GT	Correct Tracks	Misses	FP	Mismatches	MOTP	MOTA
Rouen	Distance	2627	2125	502	**519**	**19**	**0.604**	**0.604**
	Colour		2126	501	560	28	0.603	0.586
	Label		**2128**	**499**	804	143	**0.604**	0.450
Sherbrooke	Distance	4429	3029	1400	**400**	**1**	0.582	**0.593**
	Colour		**3030**	**1399**	401	6	0.582	0.592
	Label		3006	1423	503	45	**0.584**	0.555
St-Marc	Distance	8375	**6068**	**2307**	**515**	**73**	0.696	**0.654**
	Colour		6041	2334	591	93	0.696	0.640
	Label		5820	2555	1161	293	**0.700**	0.521
Rene-Levesque	Distance	9418	**2701**	**6717**	**530**	**0**	0.740	**0.231**
	Colour		2694	6724	538	15	0.741	0.227
	Label		2596	6822	687	80	**0.746**	0.194

proposed method yields better tracking performance than Urban Tracker, MKCF and Ooi et al. (Chapter 4). Overall, IMOT outperformed all evaluated methods, even though our proposed method performs the best in terms of MOTA for the video St-Marc and is second best in terms of MOTA on Sherbrooke. It is noted, however, that the proposed method gives a low MOTA for Rene-Levesque. Fusion of objects in the proposed method is not working well for this particular video as the objects in the scene are very small, and inevitably they get incorrectly paired with MOD bounding boxes that are usually large and imprecise for small objects. Consequently, this affects the overall MOT performance. In fact, Ooi et al. (Chapter 4) used only detection inputs, which was not able to track any object in this video. It was already demonstrated that the use of only MOD objects as inputs for the MOT does not work well for this particularly challenging video. The good MOTP values obtained by Ooi et al. (Chapter 4) show that MOD BBs although not very reliable can give object locations that are sometimes more precise.

Table 5.2 Comparison of the proposed method performance with state-of-the-art approaches. **Boldface** indicates best results, *italic* indicates second best.

	Our method		IMOT		Urban Tracker		MKCF		Ooi et al.	
	MOTP	MOTA	MOTP	MOTA	MOTP	MOTA	MOTP	MOTA	MOTP	MOTA
Rouen	0.604	0.601	*0.620*	*0.670*	0.617	**0.696**	0.582	0.501	**0.687**	-0.188
Sherb.	0.582	*0.595*	*0.590*	**0.690**	0.576	0.404	0.553	0.317	**0.749**	0.027
St-Marc	*0.696*	**0.654**	0.682	*0.653*	0.691	0.638	0.652	0.463	**0.723**	-0.366
Rene-L.	**0.741**	0.230	*0.705*	**0.613**	0.582	*0.565*	0.531	0.334	NA	NA

5.4.3 Discussion

The integration of objects from IMOT and a MOD is proposed in order to better capture the objects of interest during the tracking process. It was expected that the combined inputs can complement each other, producing better inputs compared to the inputs produced individually from the different approaches. For instance, with the presence of fragmented objects from background subtraction that are difficult to group together, having a reference BB from the MOD that encompasses the whole object could be a useful indicator to improve the representation of the complete object. However, from the experiments, we have noticed the tendency of the MOD to generate large BBs that often include areas that do not belong to the object of interest. While in certain frames, it is helpful to have such BBs showing objects that are partially occluded, there are many occasions that such BBs include several objects of interest as one detection, especially for objects of small sizes such as pedestrians in urban traffic scene.

This led to a difficulty of tracking them effectively as the input objects to the tracker are already merged as one whole object instead of distinct objects. In addition, there are cases where IMOT objects encompassed more than one object of interest that appeared on the scene as well due its origin of background subtraction. As an effort to mitigate these effects, we have imposed a stricter merging threshold to reduce the amount of incorrect fusion of objects. To distinguish the case between combining BBs of fragmented parts into one whole object, and the case of having multiple objects interacting in close proximity, we take into consideration the colour of IMOT objects to make the merging decision.

The excessive inclusion of areas that are not relevant may impact the tracking process as well. This is because the colour histogram will consider the background portion that was included in the BB for object description in the association cost for matching across frames, leading to possibly less accurate descriptions of the objects of interest. However, despite the effort to differentiate the two cases, some missed objects are still missed in the final tracking outputs because of the imperfect representation of some objects of interest that get fed into the tracker. The missed objects could be the result of MOD objects that are not paired with the available IMOT objects. Indeed, sometimes the MOD can detect object that IMOT cannot.

5.5 Conclusion

In this paper, we presented a novel approach for fusing input objects from a multiclass object detector and an improved object extraction approach based on background subtraction for

multiple object tracking. We use the integrated set of objects into a proposed MOT framework that associates objects across frames using spatial, colour and class label information to form trajectories in challenging urban traffic scenes. The prediction quality of unmatched objects in the MOT paradigm is evaluated to further improve the final tracking results. Results show that our method is competitive, but that it is very challenging to combine detections from multiple sources. First, they may not detect the same objects, and secondly, even if the same objects are detected, objects are not bounded in the same way. Our ablation study show that using class labels and their confidence can contribute positively to the data association cost function.

Acknowledgments

This research is funded by FRQ-NT (Grant: 2016-PR-189250) and Polytechnique Montréal PhD Fellowship. The Titan X used for this research was donated by the NVIDIA Corporation.

CHAPTER 6 ARTICLE 3: SUPERVISED AND UNSUPERVISED DETECTIONS FOR MULTIPLE OBJECT TRACKING IN TRAFFIC SCENES: A COMPARATIVE STUDY

Hui-Lee Ooi, Guillaume-Alexandre Bilodeau and Nicolas Saunier,
published at International Conference on Image Analysis and Recognition (ICIAR), March 2020

Abstract

In this paper, we propose a multiple object tracker, called MF-Tracker, that integrates multiple classical features (spatial distances and colours) and modern features (detection labels and re-identification features) in its tracking framework. Since our tracker can work with detections coming either from unsupervised and supervised object detectors, we also investigated the impact of supervised and unsupervised detection inputs in our method and for tracking road users in general. We also compared our results with existing methods that were applied on the UA-Detrac and the UrbanTracker datasets. Results show that our proposed method is performing very well in both datasets with different inputs (MOTA ranging from 0.3491 *to* 0.5805 *for unsupervised inputs on the UrbanTracker dataset and an average MOTA of* 0.7638 *for supervised inputs on the UA Detrac dataset) under different circumstances. A well-trained supervised object detector can give better results in challenging scenarios. However, in simpler scenarios, if good training data is not available, unsupervised method can perform well and can be a good alternative.*

6.1 Introduction

Multiple object tracking (MOT) in the context of traffic scenes essentially means following the target objects (road users) in the scene to obtain an accurate representation of their trajectories across frames, usually as feedback information to eventually improve traffic management systems or to better plan the layout of the roads. To follow an object, we must see it first; to track a road user in a scene, the importance of getting correct detection inputs for the tracking paradigm must not be overlooked. Compared to single object tracking, MOT has to keep track of the presence of more than one target object while dealing with the possible occlusions and mismatches of objects as a result of interactions of the moving objects with the background and other objects, making it a challenging problem that is still actively researched. In the case of traffic scenes, the MOT method must also deal with various lighting

and weather conditions (See Figure 6.1). There are also multiple classes of objects.

Generally, there are two types of object detection methods to be used for tracking: supervised and unsupervised. The former is the more modern approach using labeled data to train models that can detect the target objects in a particular domain [79, 114]. This approach usually delineates an object with a bounding box, and also attributes a class label to each detected object. The latter typically corresponds to the classical approach of foreground extraction and outputs objects that are not part of the background in the frame [115, 116]. This method does not need supervised training as it segments the scene in two classes based on a model of the background. It is designed for cameras that are not moving and provides an object segmentation mask, but no labels.

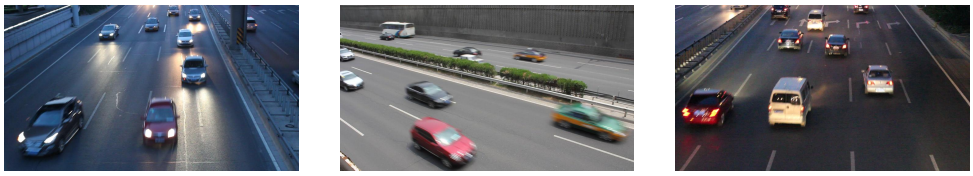
In this paper, we address the MOT problem for traffic scenes by proposing a new tracker that integrates classical features (spatial distances and colours) and modern features (detection labels and re-identification features), as well as object prediction in its tracking framework. Our tracker can be applied to either supervised and unsupervised object detections. Therefore, while designing our method we raised the question: *which type of detection should be used?* To answer this question, we investigated formally the impact of the choice of the type of detections in the design of the tracker.

The contributions of this paper are: 1) a new MOT tracker that combines various features and that can capitalize on both unsupervised and supervised object detections and 2) a formal analysis of the performance of unsupervised and supervised object detectors in road tracking scenarios and their impact on MOT.

6.2 Related Works

The study of MOT on traffic scenes has undergone many changes and evolution over the years. Conventionally, before the advent of deep learning in computer vision, the extraction of target objects in the application of MOT were generic and unsupervised, as in [20, 34, 117]. In [34], background subtraction detections were combined with kernelized correlation filters (KCF) for solving the MOT problem in urban traffic scenes. KCF is used as an appearance model as well as for predicting the object position in the next frame. Similarly, [20] used background subtraction to extract potential unknown road users for their proposed finite state machine to handle the different target objects. Keypoints are used as an appearance model. Other works, like the one of [32], instead used optical flow information to detect and track road users.

Recently, most works on MOT use detections from supervised learning methods that output



(a) (b) (c)

Figure 6.1 Examples of selected frames from videos in the UA-Detrac dataset [3] used for evaluation in the experiments.

bounding boxes around learned object classes. The use of a deep learning-based detector as the only source of input for multiple object tracking involving several different types of road users was presented in Ooi et al. (Chapter 4), but with disappointing results. Ooi et al. (Chapter 5) then further improved the method on the same dataset (UrbanTracker [20]) by applying classical unsupervised object detection outputs coupled with modern supervised learning-based detector outputs, achieving some progress with the use of detector labels as part of the feature description as well.

Meanwhile, the reported results on the UA-Detrac dataset [3] on its official website are based on supervised object detectors. UA-Detrac does not consider bikes, motorcycles and pedestrians. At the time of conducting our experiments, the reported top trackers on the dataset are Evolving Boxes (EB)+Kalman+IOUT (extension of [52]), EB+IOUT [52] and RCNN+IOUT [52]. These three methods are rather similar, essentially working by the overlap of the intersection over union (IOU) of the bounding boxes that represent the objects in each frame, with the assumptions that the high frame rate of the videos does not leave "gaps" between the detections [52]. The Kalman filter used in the EB+Kalman+IOUT approach is meant to allow skipping frames via predictions to improve processing speed.

Hence, in this study, we are interested in evaluating and understanding the effects of unsupervised and supervised detections for MOT in varying traffic scenarios under different environmental conditions as provided by these two datasets, UA-Detrac and UrbanTracker. We therefore devised a novel tracker that can work with both kinds of inputs.

6.3 Method

We proposed a novel tracker (MF-Tracker) that combines classical features as well as deep learning features for the matching of objects across frames. We are also interested in investigating the effects of supervised and unsupervised detections on MOT performance. Our tracker was thus designed to work with both types of detections.

Our multiple object tracker consists of several components: (i) Object detection, (ii) Feature generation from objects and (iii) Data association to produce the final tracking outputs that describe the trajectory of each target object across frames, as shown in Figure 6.2.

6.3.1 Inputs for the Tracker

Since we intend to compare the performances of different input objects for tracking, we used a state-of-the-art background subtraction method (IMOT [35] with PAWCS (Pixel-based Adaptive Word Consensus Segmenter) [116]) as unsupervised input source and the deep learning-based detector (RetinaNet [79]) as supervised input source. Both approaches give bounding boxes of target objects for each frame.

The next step for MF-Tracker is to extract the information contained within the bounding boxes for the subsequent tracking module.

6.3.2 Classical Features and Modern Features

The proposed method integrates both classical features and modern features to generate overall similarity scores to compare the objects across frames.

The similarity costs from classical features are:

- the spatial cost C_d: based on the spatial distances of the four coordinates of the bounding boxes, it is defined as:

$$C_d = 1 - max(0, \frac{T_d - \overline{SD}}{T_d}) \tag{6.1}$$

$$\overline{SD} = \frac{1}{4}(|x_{D,min} - x_{T,min}| + |y_{D,min} - y_{T,min}| + |x_{D,max} - x_{T,max}| + |y_{D,max} - y_{T,max}|), \tag{6.2}$$

  where $\overline{SD}$ is the mean bounding box spatial distance and x_{min}, y_{min}, x_{max} and y_{max} denote the minimum and maximum coordinates of an object bounding box. T represents an object that is currently tracked while D represents a detected object in a frame. $\overline{SD}$ denotes the mean spatial distance of the x coordinates and y coordinates of all the four corners of the bounding boxes of the compared objects. A fixed parameter T_d is used to normalize C_d and to bound the maximal distance between bounding boxes.

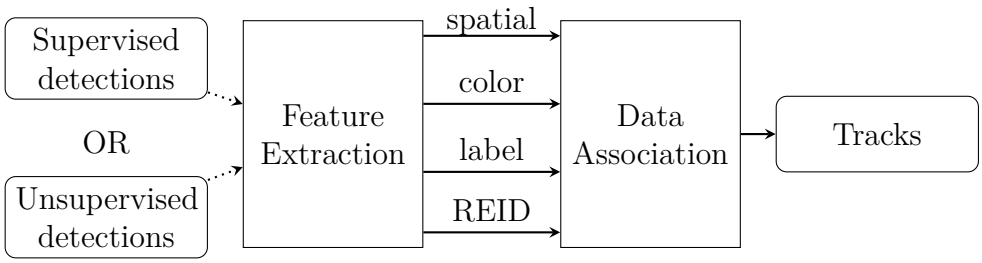


Figure 6.2 Overview of our proposed tracker (MF-Tracker). Detections from supervised or unsupervised approaches are fed into the Feature Extraction module for further processing in Data Association to produce the final trajectory outputs.

- the color cost C_c: it is the Bhattacharyya distances of the color histograms of the bounding boxes. It is defined as:

$$C_c = \sqrt{1 - \frac{1}{\sqrt{\overline{H_i^D}\,\overline{H_j^T}\,N^2}} \sum_N \sqrt{H_i^D H_j^T}}, \tag{6.3}$$

  where H_i^D denotes the color histogram of a detection i, H_j^T denotes the color histogram of a currently tracked object j and N is the total number of histogram bins (256 is used in this work). $\overline{H_m^D}$ and $\overline{H_m^T}$ are the histogram bin means of the detected object and currently tracked object, given by Equation 6.4 and Equation 6.5 respectively.

$$\overline{H_m^D} = \frac{1}{N} \sum H_m^D \tag{6.4}$$

$$\overline{H_m^T} = \frac{1}{N} \sum H_m^T \tag{6.5}$$

Meanwhile, the similarity costs from modern features are:

- the label cost C_l: the label information from the detector inputs are used as a similarity cost. It is defined as:

$$C_l = \begin{cases} 1 - \frac{W_i + W_j}{2} & \text{if } L_i = L_j \\ 1 & \text{if } L_i \neq L_j, \end{cases} \tag{6.6}$$

  where L_i denotes the class label of object i and W_i its confidence value (between 0 and 1). Using the confidence value from the object class label, and not just the class label for the cost is a beneficial strategy because confidence values tend to be similar in consecutive frames for a given object.

- the re-identification (REID) cost C_r: the deep-learned REID features of OSNet [118] are also used for object description, where the REID cost is computed with the Euclidean distance as

$$C_r = 1 - \sqrt{\sum_n (r_n^i - r_n^j)^2} \tag{6.7}$$

  where r_n^i and r_n^j denote respectively the n_{th} REID feature value of object i and j, and n is the number of REID features. We used OSNet pre-trained on Multi-Scene Multi-Time person ReID dataset (MSMT17) [119]. The features were not specifically tuned for our application.

All these features are applied and combined to give a final similarity score given by

$$C_{final} = \alpha C_d + \beta C_c + \gamma C_l + \lambda C_r, \tag{6.8}$$

that ranges from 0 to 1, and where $\alpha, \beta, \gamma, \lambda$ denotes the weights for the corresponding cost.

This procedure is performed in the extracted bounding boxes of detections from both supervised and unsupervised sources. In the experiment, for the case of unsupervised detections, due to lack of label information from the unsupervised method itself, detections from the supervised detector are matched with the ones from the unsupervised approach, thus assigning the label accordingly to the bounding boxes given by the unsupervised detector. Alternatively, an object classifier could be applied. Input detection boxes from the unsupervised approach are given null labels if there is no overlapping boxes from the supervised approach.

6.3.3 Data Association

Based on the similarity score computed from the features, the Hungarian algorithm is used for matching the detected objects (detection list) from the supervised or unsupervised approaches in each frame to the tracked objects (tracked list) accumulated from the previous frames.

Corresponding objects from the two lists (detection list and tracked list) are marked as matched detection and the information for the objects is updated accordingly. Objects from the detection list that are not successfully matched with any object in the tracked list are initialized as new objects and taken in as part of the tracked list for the subsequent frame. Unmatched objects from the tracked list could either be objects that are occluded or objects that already left the scenes, or invalid objects that are incorrectly detected. A Kalman filter is used to make prediction in the subsequent frames, accounting for occlusion cases, so that occluded objects in the tracked list proceed with possible trajectories when they were momentarily not detected at certain frames.

For each object trajectory, there is also an analysis on the position histories so as to remove invalid objects that are not relevant or to terminate the trajectory when the objects were confirmed to have left the scene.

6.4 Experiments

The UA-Detrac [3] and UrbanTracker [20] datasets were used for the evaluation in this study because they include four challenging real-world traffic videos with 4 to 20 targets in the same frame simultaneously under different environmental conditions with varying types of annotated road users. The videos contain 600 to 1000 frames respectively. Evaluation of performances for the two datasets is performed using the standardized CLEAR MOT metrics [91]. Because unsupervised detections are less precise in their localization and extent (see Figure 6.3), an intersection over union of 0.3 is used for computing the evaluation metrics as in previous work [35].

6.4.1 Experimental setup for the UA-Detrac dataset

Comparing supervised and unsupervised detections is not trivial because datasets are designed with one or the other in mind. UA-Detrac does not include annotations for pedestrians, bikes and motorcycles. Due to the nature of unsupervised methods in producing the input objects for the tracker, it is observed that the presence of these unannotated road users in the frames will severely affect the quality of inputs for tracking and perhaps good trajectories without corresponding annotations will be produced, but penalized in the MOTA. Hence, in order to allow for fair comparisons of performance for the two sources of inputs in the tracking phase, we have chosen 22 videos from the training set for this evaluation where there are no (or very few) pedestrians, bikes and motorcycles. The videos are recorded at 25 fps (frame per seconds) with resolution of 960x540 pixels. The chosen videos include different angles of observations with varying illumination and weather conditions. Comparison of existing methods on the dataset is done by running the trackers on these videos individually to obtain their MOTA and MOTP results.

For an unsupervised method, to get the detections, the background subtraction method typically observes the video for some time to learn the background. In UA-Detrac, objects have to be detected and tracked from the first frame of the video. Therefore, to simulate the normal condition in which an unsupervised method would be applied, for each selected video, k frames are selected randomly over the whole video for learning the background. That way, foreground objects can then be detected from frame 1 in the tracking evaluation. Hence

Figure 6.3 Examples of extracted bounding boxes from supervised and unsupervised detections of road users in evaluated sequence.

practically, for the unsupervised approach to work on this dataset, it has to "see" certain portion of the frames of the video before doing the actual foreground detection. Hence to allow fair comparison with the supervised methods, we are conducting experiment on videos of the training set, where the detector has "seen" the data as well.

In practical applications where the evaluation is performed on new unseen data, it is expected that the tracking performance will be lower for both type of detectors because of some deterioration in quality of the detections obtained.

For the supervised detections as used in our method, RetinaNet with VGG-16 as backbone is trained on the training set of UA-Detrac. The detected objects with confidence lower than 0.4 are filtered out before tracking. As for unsupervised detections, only bounding boxes with areas that are greater than 2000 pixels are allowed as input to the tracker. These steps are to ensure that only input objects that are valid in terms of size and confidence will be used for our MOT evaluation. Indeed, presence of spurious noise and incorrect detections can have a detrimental effect on the overall tracking performances. The supervised and unsupervised detections used in our experiments with UA-Detrac can be downloaded at this link (`https://github.com/HuiLee-Ooi/MF-Tracker`).

Besides comparing results of our proposed tracker with the different detection sources, tracking performances of existing trackers, under similar experimental settings with supervised and unsupervised detections, are presented as well in Table 6.1.

Table 6.1 Comparison of MOTA and MOTP performances of trackers with supervised and unsupervised detections on selected videos of UA-Detrac. For tracker names, the part following "+" indicates the method used to obtain detections. **Boldface** indicates best result, Underline indicates second best result and *Italicized green* indicates third best result.

	Unsupervised detections								Supervised detections					
Video Seq.	MKCF + ViBe		IMOT + ViBe		MKCF + PAWCS		MF-Tracker+ IMOT-PAWCS		IoU + EB		IoU + RCNN		MF-Tracker + RetinaNet	
	MOTA	MOTP	MOTA	MOTP	MOTA	MOTP	MOTA	MOTP	MOTA	MOTP	MOTA	MOTP	MOTA	MOTP
MVI_39801	-1.1280	0.5624	-1.1493	0.5301	0.1309	0.5399	0.1970	0.5859	*0.6085*	0.8146	0.6773	*0.7485*	**0.8351**	**0.8536**
MVI_39861	-2.3416	0.5928	-2.0680	0.5319	-0.7905	0.5392	-0.0244	0.6201	0.7529	0.8423	*0.5502*	*0.7312*	**0.7824**	**0.8546**
MVI_40191	-1.1280	0.5624	-0.6120	0.6227	0.1679	0.6001	0.3050	0.7227	0.7201	0.8979	*0.5156*	*0.8337*	**0.8549**	**0.9123**
MVI_40192	-1.6718	0.5357	-1.0181	0.5948	0.3896	0.6055	0.3615	0.6915	0.5273	0.8807	*0.4574*	*0.8145*	**0.7999**	**0.8918**
MVI_40201	-2.4787	0.5489	-0.7528	0.5861	0.4245	0.6182	0.3321	0.6687	0.4643	**0.8897**	0.6324	*0.8168*	**0.8009**	0.8873
MVI_40204	-0.9831	0.5456	-0.7504	0.5700	0.2506	0.6050	0.2368	0.6651	**0.7799**	0.8645	0.6676	*0.7647*	0.5312	**0.8715**
MVI_40211	-5.1159	0.6144	-3.0001	0.5892	0.1305	0.6642	0.3414	0.6514	**0.8491**	0.9011	*0.6354*	*0.7703*	0.7019	**0.9017**
MVI_40212	-3.3782	0.6059	-1.9731	0.5858	0.0832	0.6576	0.2650	0.6438	**0.8446**	**0.8952**	*0.6485*	*0.7731*	0.7452	0.8841
MVI_40213	-3.1261	0.5969	-1.7845	0.5928	0.2929	0.6594	0.3957	0.6430	**0.8458**	**0.9023**	*0.5389*	*0.7699*	0.7028	0.8920
MVI_40241	-0.5776	0.5802	-0.3539	0.6214	0.3978	0.6246	*0.4493*	0.6880	0.3936	0.8998	0.6279	*0.7821*	**0.7535**	**0.9116**
MVI_40243	-0.0120	0.5995	-0.0934	0.6424	0.3950	0.6177	*0.4828*	0.6862	0.2845	0.9009	0.5216	*0.7860*	**0.7695**	**0.9116**
MVI_40244	0.1491	0.5771	0.0872	0.6458	0.4985	0.6091	*0.5448*	0.6770	0.1784	0.9044	0.5818	*0.7843*	**0.7316**	**0.9139**
MVI_40752	-0.5572	0.5888	-0.3374	0.6301	-0.0946	0.5952	0.2330	0.6586	0.6464	**0.8799**	*0.5782*	*0.7521*	**0.7607**	0.8788
MVI_40871	-0.4175	0.4671	-0.5729	0.4176	-0.1294	0.5151	0.0418	0.5522	*0.1642*	0.8861	0.4538	*0.8061*	**0.8208**	**0.9208**
MVI_40962	-0.4033	0.5940	0.0078	0.6796	-0.2181	0.5943	0.3114	0.7298	*0.6969*	0.9140	0.8478	*0.8488*	**0.8696**	**0.9240**
MVI_40963	-0.3398	0.5442	-0.0243	0.6428	-0.1518	0.5237	0.2407	0.6730	**0.7308**	**0.8637**	0.7056	*0.7699*	*0.5945*	0.8441
MVI_40981	-0.6972	0.5346	-0.4488	0.5951	-2.0140	0.4758	0.0065	0.6000	*0.7529*	0.8915	0.8122	*0.7932*	**0.8964**	**0.9168**
MVI_41063	0.1081	0.6198	0.1116	0.6271	0.4788	0.6399	0.3598	0.6507	0.7666	0.8738	*0.7283*	*0.7868*	**0.7939**	**0.8870**
MVI_41073	-1.1475	0.6197	-0.6290	0.6440	-0.4349	0.6355	0.2526	0.6820	**0.8098**	**0.8889**	0.7954	*0.7699*	*0.7320*	0.8732
MVI_63552	-3.2291	0.5804	-2.2304	0.5329	-0.1617	0.6421	0.1696	0.6249	0.6236	0.8334	*0.5364*	*0.7487*	**0.7518**	**0.8549**
MVI_63553	-3.0276	0.5699	-1.5817	0.5530	-0.1176	0.6256	0.1720	0.6161	0.7878	0.8455	*0.5474*	*0.7433*	**0.8032**	**0.8499**
MVI_63554	-3.0283	0.5657	-1.9908	0.5631	0.0836	0.6122	0.2065	0.6278	0.7213	**0.8739**	*0.5668*	*0.7698*	**0.7707**	0.8660
average	-1.5696	0.5730	-0.9620	0.5908	-0.0177	0.6000	0.2673	0.6527	0.6341	0.8793	*0.6194*	*0.7802*	**0.7638**	**0.8864**

At the time of writing, the current reported three best trackers in the dataset official website are based on [52] with detection results from [114] and [120]. However, since we are not able to run the tracker version with the Kalman filter on the individual videos presented in this study (the public code does not work), only results of EB+IOU and RCNN+IOU are reported (as IoU + EB and IoU + RCNN in the table).

6.4.2 Experimental setup for the UrbanTracker dataset

In this experiment, all four videos in the UrbanTracker dataset are used to evaluate and compare with existing methods.

The optimal filter for the size of detections varies depending on video due to the inherently different scenarios. For a fair comparison, we are using the same parameter settings as presented by [20]. Meanwhile, due to the limited amount of data in the dataset, supervised detection inputs are results of RetinaNet detection with VGG-16 backbone trained on the UA-Detrac training set. The confidence threshold for filtering out the input bounding boxes from supervised sources is set at 0.4 for all videos. For unsupervised detections, extra frames are available before the annotated video segments. They are thus used to learn the background model.

The MOT performances for our proposed MF-Tracker (with supervised and unsupervised detections) compared to existing methods are presented in Table 6.2.

Table 6.2 Comparison of MOTA and MOTP performances of trackers on the UrbanTracker dataset. For tracker names, the part following "+" indicates the method used to obtain the detections. **Boldface** indicates the best result, Underline indicates the second best result and *Italicized green* indicates the third best result. * indicates that the reported results are taken from original published works without re-running the methods. RL indicates Rene-Levesque and Sher. indicates Sherbrooke

Video Seq.	Unsupervised detections						Supervised detections			
	MF-Tracker + IMOT-PAWCS		UrbanTracker + IMOT * [35]		MKCF + ViBe * [34]		MF-Tracker + RetinaNet		Ooi et al. (Chapter 4) + RFCN [92]	
	MOTA	MOTP	MOTA	MOTP	MOTA	MOTP	MOTA	MOTP	MOTA	MOTP
Rouen	0.5805	*0.6035*	**0.670**	0.620	*0.501*	0.582	0.133	0.885	-0.188	**0.687**
Sher.	0.609	0.5771	**0.690**	*0.590*	0.317	0.553	*0.3771*	**0.915**	0.027	0.7490
St-Marc	0.643	*0.6849*	**0.653**	0.682	*0.463*	0.652	0.1124	**0.951**	-0.366	0.723
RL	0.3491	0.712	**0.613**	*0.705*	*0.334*	0.531	0.273	**0.901**	NA	NA

6.5 Results

For the UA-Detrac dataset, generally, the trackers with supervised detections give better tracking performances than the ones with unsupervised detections, as shown in Table 6.1.

MF-Tracker outperformed all the compared methods when coupled with supervised detections.

Supervised detections on the UA-Detrac dataset work very well, where MF-Tracker + RetinaNet achieved a mean MOTA of 0.7638 and a mean MOTP of 0.8884, whereas unsupervised detections are not as good with MF-Tracker + IMOT-PAWCS only achieving mean MOTA of 0.2673 and mean MOTP of 0.6527, despite the use of a state-of-the-art background subtraction method.

Despite the trend of supervised detectors overwhelmingly giving better performances than unsupervised detectors, it is interesting to note that for some videos (MVI_40241, MVI_40243 and MVI_40244), our tracker with unsupervised detections ranked in third place, being quite competitive with the second ranked method (IoU + RCNN) that is based on supervised detections. These three videos are observed to have fast vehicles moving, causing motion blur. It is also observed that the use of state-of-the-art background subtraction (PAWCS [116]) with MKCF (Multiple Kernelized Correlation Filters) improves the performance of the original MKCF that uses ViBe background subtraction. Similarly, the use of PAWCS [116] with IMOT with our tracker has improved the tracking performance compared to the original implementation of the IMOT approach based on ViBe [115]. IMOT post-process results from background subtraction by using optical flow and edges to solve object merging.

On the contrary, the results on the UrbanTracker dataset are showing a different trend. Table 6.2 shows that trackers with unsupervised detections give better performances in terms of multiple object tracking accuracy (MOTA), especially UrbanTracker + IMOT [35]. Our proposed tracker with unsupervised detections (IMOT boxes from PAWCS background subtraction) ranked second in the comparison. However, it must be noted that the results reported in both [35] and [34] are using parameters that are specifically tuned to each video in the dataset. In contrast, aside from the filter for input size in the tracker that varies according to video (which is a useful step given the disparity of target input size among the videos and because those filter sizes were used by the competing methods), the proposed MF-Tracker is applied with identical parameter settings for all the evaluated videos in the dataset. Still, MF-Tracker with unsupervised detections obtains competitive results with respect to [34, 35] for Rouen, Sherbrooke and St-Marc, although tracking performance on Rene-Levesque is significantly worse.

Table 6.3 Comparison of tracking results on videos from UrbanTracker dataset based on the different individual features

Features	Correct Tracks	Misses	FP	Mismatches	MOTP	MOTA
distance	19358	5491	5182	89	0.677	0.567
color	19292	5557	5371	141	0.677	0.555
label	18968	5881	5193	271	0.679	0.543
REID	19090	5759	6761	654	0.678	0.470

Effects of the four different feature cost on our proposed tracker were studied individually on the UrbanTraccker dataset with supervised inputs in Table 6.3, where number of correct tracks, misses, false positives (FP) and mismatches of the four videos are accumulated. It is observed that the compared features gives fairly similar MOTP (0.68) and MOTA ranges from 0.47 to 0.57. Spatial distance appears to be the best performing feature whereas REID is the worst performing feature. Therefore we used the following weights $\alpha = 0.7, \beta = 0.1, \gamma = 0.1$ and $\lambda = 0.1$ in the experiments.

6.6 Discussion

The quick impression from the presented results is that supervised methods give better detections for the UA-Detrac dataset and conversely, unsupervised detections work better on the UrbanTracker dataset.

For UA-Detrac, while the use of state-of-the-art background subtraction might help improving the tracking results (comparing original MKCF with ViBe and MKCF with PAWCS), it is obvious that the methods with supervised detections are the clear winners. While one could argue that good results are expected since the videos are part of the training set, similar conditions can be said on the unsupervised methods as well since each video are "seen" to build the background model before producing foreground outputs (detections) for tracking purposes. However, despite a similar amount of learning on the data itself, methods with unsupervised detections with fixed parameter settings still yield poor results overall.

Unsupervised object detection methods struggle with high density traffic were all objects become merged together. Supervised object detection methods handle these cases better because each road user is individually detected. Also, for unsupervised detection methods, in night conditions, car headlights generate foreground regions that are then tracked as ghost objects. They are ignored by supervised detection methods.

In any case, our proposed tracker with unsupervised detections (MF-Tracker + IMOT-PAWCS) is the best performing method among the methods with such detections, and it

managed to rank third on three of the videos in terms of MOTA, effectively outperforming a method with supervised detections (IoU + EB). These videos are revealed to be containing high speed vehicles that appear slightly blurry in the frame, possibly causing the supervised detector to produce less accurate detections for the tracking framework. On the other hand, the camera that is statically positioned ensure that the backgrounds of the videos are properly learned without a lot of noise by the unsupervised detector, thereby producing detections of satisfactory quality to proceed with tracking. It must be noted that while the videos in the UA-Detrac dataset are taken from fixed camera setups, some inevitable environmental conditions such as windy weather can affect the quality of foreground given by unsupervised detectors as the camera is slightly moving and vibrating. In these cases, results show that newer methods (e.g. PAWCS) can better handle this issue than older methods (e.g. ViBe), where street markings and highways dividers are detected as objects.

As we delve deeper in interpreting the results, it is observed that the supervised detectors do not perform as well on the UrbanTracker dataset as on the UA-Detrac dataset because the datasets contain inherently very different scenarios. The UA-Detrac dataset contains a large number of videos in similar locations and angles with subtle differences, such as illumination at different time of day. In contrast, the four videos in the UrbanTracker dataset are captured at entirely different locations and the different heights of installation of the cameras cause the captured objects in the frames to be highly varied in sizes and scales. UrbanTracker also contains a larger variety of viewpoints. The work of Chapter 4 on UrbanTracker dataset has previously shown that a supervised detector performed poorly on the dataset, due to detector that produces too many false positive objects for tracking. Both MF-Tracker + RetinaNet and the tracker presented in Chapter 4 are not trained on UrbanTracker itself due to the lack of available training videos. It is plausible that better results could be achieved by supervised detectors with more relevant training data, which is unfortunately lacking for proper training.

The best performances on the UrbanTracker dataset are from UrbanTracker + IMOT [35], while our proposed tracker with unsupervised detections ranked second in terms of MOTA for all the videos. However, aside from the size filter for the unsupervised detections to be fed into the tracker, our proposed tracker retained all the same parameters and settings for all the videos. This is not the case for the tracker parameters in the works of [35] and [34] that have been tuned to each of the specific videos in the dataset to achieve competitive final results. It is important to note that this could be the main reason why UrbanTracker + IMOT generally fare better on the UrbanTracker dataset. In practical real applications, however, it is desirable to have generic settings that is not overly tuned (overfit) to individual video sequence.

6.7 Conclusion

We presented a novel multi-feature tracker (MF-Tracker) that comprises classical and modern features for the matching of objects across frames. In addition, we evaluated our tracker with either unsupervised or supervised object detection approaches to investigate their differences in MOT performance. Compared to the existing trackers evaluated on the datasets, our proposed tracker achieved the best performances on the UA-Detrac dataset and is highly competitive on the UrbanTracker dataset with fixed parameters for all videos during tracking. Supervised inputs, when sufficiently trained with available data, produce good inputs that lead to more accurate tracking of objects. Nevertheless, in simpler scenarios, if good training data is not available, unsupervised method can perform well and can be a good alternative that should not be neglected.

Acknowledgments

This research is funded by FRQ-NT (Grant: 2016-PR- 189250) and Polytechnique Montréal PhD Fellowship. The Titan X used for this research was donated by the NVIDIA Corporation. We acknowledge the contribution of Hughes Perreault for providing the RetinaNet detections.

CHAPTER 7 GENERAL DISCUSSION

This chapter discusses the thought process for the different attempts and experiments that lead to the eventual ideas implemented in the published papers of Chapter 4, 5 and 6. With these additional results and unpublished attempts at the initial stage of our study, the observations and results gained throughout the research process have propelled the direction of the proposed MOT approach into its current state.

At the earliest stage of the project, the work was mainly focused on obtaining targets of good quality from the scene prior to proceeding to the eventual tracking task.

7.1 Object detection with superpixels

Some earlier experiments were conducted in applying superpixel processing on the frames to obtain targets in the form of superpixels in order to better capture their characteristics and possibly detect them more precisely in a frame. Various types of superpixel approaches were experimented such as SLICs [121] (see Figure 7.1) and SEEDS [122] with varying parameters to find the optimal settings for better results. This approach was envisioned to obtain a good extraction of foreground superpixels that represent targets with respect to the background superpixels for the MOT application. That is, to improve foreground detection, information from the superpixels is used to better localize the boundary of the objects. For example, to remove shadows as depicted in Figure 7.1b.

However, there are occasions when some superpixels contain some portions of the targets as well as the background, which is irrelevant for our tracking purposes. The challenge was to perform a clean extraction of superpixels for the road users, especially around the boundaries of the targets from its surrounding environment.

Experiments were performed to evaluate the combination of background subtraction and superpixel segmentation to identify foreground superpixels from the background superpixels. Since background subtraction will give an indication on which pixels belong to the background or the foreground, we were interested in extracting superpixels that are identified as foreground, tentatively setting them as the targets to be tracked by observing the pixels contained by the superpixel. It was postulated that the superpixel entity will give a better representation without the noisy individual pixels that might interfere with the tracking process.

One of the initial ideas was to count the number of foreground pixels in a superpixel and apply

(a) (b)

Figure 7.1 Background subtraction implemented on a video frame in (a) pixel-level and (b) SLIC superpixel-level with compactness = 10 and size = 20.

some majority voting mechanism to label the whole superpixel as foreground or background (see Figure 7.1). Although this approach could remove noisy foreground pixels, it was also often extending the object area by adding additional boundary pixels. In addition, labeling decisions were difficult in the case where there was almost as many background pixels as foreground pixels inside a superpixel.

At the beginning of this thesis, another initial idea was to identify the different targets of interest in the scene by observing the motion of the pixels contained by the superpixels. Since there are some regions of the targets that are missed out by background subtraction, we had the idea of integrating the motion information as well to better extract the foreground superpixels. Optical flow is combined with the superpixels (as shown in Figure 7.2) in order to find the motion information in terms of flow magnitude and direction for each superpixel.

One of the proposed attempts was to categorize a superpixel entity as foreground or background by evaluating its overall direction and magnitude of motion obtained from the optical flow information on the clusters of pixels since theoretically, different types of road user (cars vs pedestrians) move at different speed and direction. This would have allowed us to solve occlusions between two objects. Experimentation with the binning of motion was performed according to the flow angle with regards to the direction to quantify the superpixels into different categories. The binning of motion direction was performed in both North-South direction and West-East direction to ensure that the motion is adequately described (see Figure 7.3). We wanted to use a limited number of bins. As shown in Figure 7.3a, the motion similarity in the N-S direction can be quantified with four bins, but it is unable to distinguish the motion pattern in E-W (motion direction from left to right and motion di-

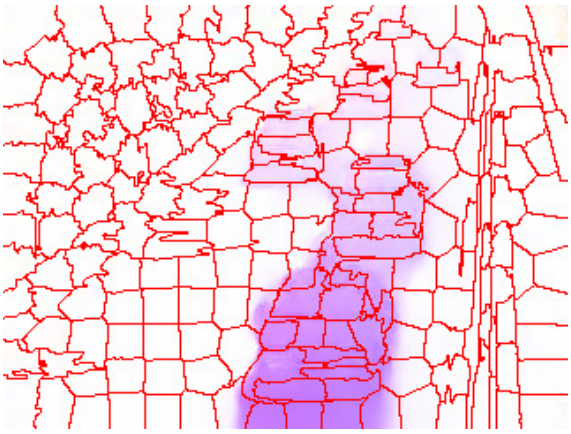
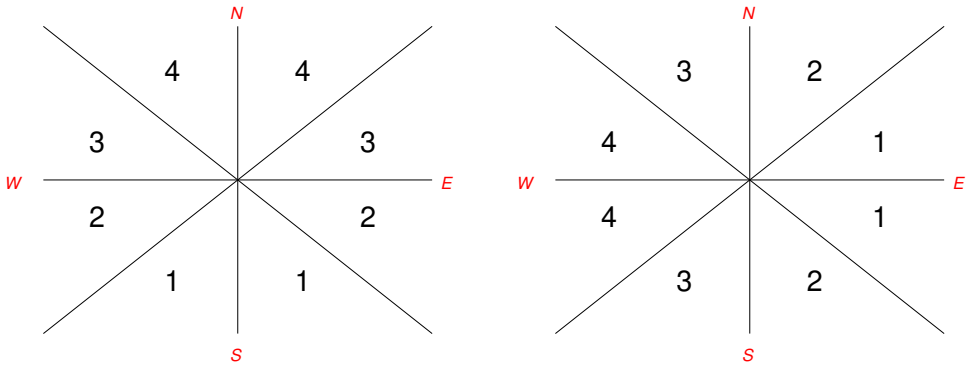


Figure 7.2 Magnitude of flow field with superpixel on video frame using Brox optical flow [4], where darker color indicating larger flow magnitude.

rection from right to left are represented in the same bin). Hence the other binning process (Figure 7.3b) is proposed to include such distinction. Note that the use of only the binning process at Figure 7.3b would have similar weakness, which is its insensitivity towards a motion pattern in N-S direction. Then, neighboring superpixels belonging to the same category (using the bin values) could be merged to form a single object, thereby addressing the occlusion problem.



(a) a (b) b

Figure 7.3 Motion bin assignment in two-step binning process on the flow angle with emphasize on (a) NS direction and (b) WE direction.

Nonetheless, the caveat is that two close angles might fall into different bins due to the quantization. We found that the representation by superpixels is just as challenging as the conventional pixel representation when it comes to merging superpixels that represent the same target. It was hypothesized that the motion information within the superpixel would be

able to give a more generalized and salient information while disregarding some of the noisy pixels that were part of the superpixel region. The evaluated results from the experiments, however, showed that it was not a feasible solution since the boundary superpixels (foreground superpixels as well as background superpixels) were indistinguishable due to the similarity in terms of optical flow information, as illustrated in Figure 7.4. We were not able to obtain better boundaries that separate the foreground superpixels from the background superpixels. The experiment has shown that the resulting extraction is highly noisy.

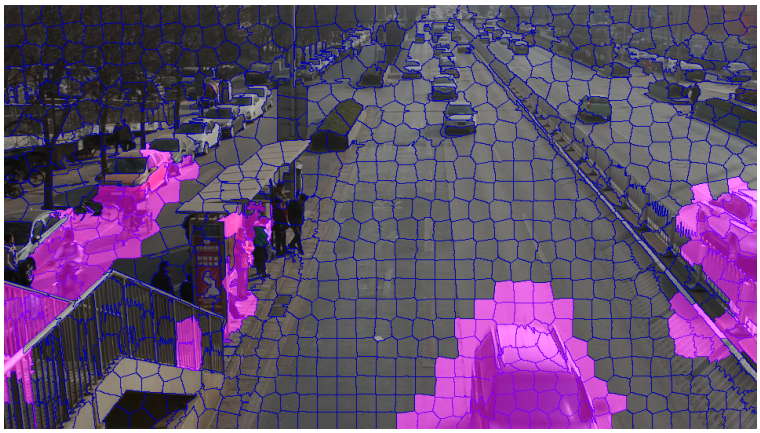


Figure 7.4 Example of attempted foreground superpixel extraction from the background based on the optical flow information.

Since it was concluded that the representation in superpixels does not give a noticeable advantage compared to the conventional pixel representation or the bounding boxes that were commonly used by learned object detector, the idea of using superpixels as a part-based model for our proposed tracking paradigm was eventually discarded.

7.2 Object detection with deep learning

Next, a deep learning-based method was employed to extract targets from the scene for our tracking framework, given that there was a growing popularity with potentially good detection results being reported in the recent literature. Compared to conventional object extraction methods, such approaches require intensive training on several datasets as well as fine-tuning on the target dataset to produce satisfactory detection results that are usable for the tracking stage.

Our first paper (Chapter 4) used a multi-class object detector called Region-based Fully Convolutional Network (RFCN) [92] that was pre-trained and fine-tuned for the application of urban traffic scenes with the MIO-TCD dataset [103] in order to obtain targets with

different categories. At the time of experimentation, the RFCN detector was chosen due to it being a SOTA approach for object detection with very good reported results. Due to the size limitation of the Urban Tracker dataset, there was a need to perform transfer learning from another dataset. The RFCN could not be fine-tuned on the Urban Tracker dataset. MIO-TCD that contains road users so that the model is capable of producing detection of targets of satisfactory quality for our urban MOT application filled this need. However, a substantial number of redundancies are observed in the resulting extraction of objects. A non-maximal suppression (NMS) method [104, 105] was applied as an attempt to reduce the number of bounding boxes that represent the same objects of interest in the video frames. It was able to eliminate some redundancies to a certain degree but it was not always effective for all cases and scenarios. It was also observed that the road users that are small in size and appeared to be far in the scene are not easy to detect with this approach. In the evaluated Urban Tracker dataset, one particular video (René-Lévesque) was having severe problems with the extraction of useful inputs for our tracking paradigm because all the road users were small. Without domain adaptation, RFCN was struggling on the Urban Tracker dataset, and results were actually worse than with background subtraction.

At this point, with the detections from RFCN as inputs, our tracking approach was somewhat rudimentary as association of objects is performed solely based on the Hungarian algorithm with predictions from a Kalman filter, as inspired by previous works, such as [22]. The Kalman filter was used for objects that are temporarily missing due to possible occlusions. In addition to the standard features used for describing the objects like positional distance, and differences of color histograms, the use of a class label distance was proposed, formulating the relationship of the compared objects such that lower matching cost would be assigned if they are of the same class label and with high detection confidence. Higher cost would be given if the compared objects have different labels or having low confidence score for their labels. It was demonstrated that the proposed label class feature is useful in boosting the performance of tracker through the ablation studies.

It was observed that the quality of detection critically affects the tracking performance on the different road users. On one hand, missing detections on the targets could lead to inaccurate tracking. On the other hand, having the redundant objects creates more confusion during the tracking process as well due to the false detections of targets. This problem was encountered during the development of Chapter 4 and Chapter 5, where this issue is highlighted and discussed in the articles as well.

Results from Chapter 4 have cemented the idea that the detection quality will greatly improve or severely degrade the final outputs of the tracker. This spurred us to work harder on the

effort on extracting foreground of high quality in our work.

7.3 Combining objects from deep learning and background subtraction

Our next attempt involved the exploration of different methods of foreground extraction to eventually obtain the most optimal inputs available for the tracking process. This task was later discovered to be more difficult than expected since simple merging of bounding boxes is insufficient to give optimal results that lead to better tracking performance. Indeed, if two detection sources give contradictory results, which one is right? This eventually led to the idea of fusing the bounding boxes from a learned object detector and a background subtraction-based approach to verify and remove the redundant bounding boxes given by the multi-class detector. Using the same detector (RFCN) with the same pre-trained steps as in Chapter 4, Chapter 5 follows the exact same approach to obtain supervised inputs in the form of bounding boxes. The same settings are kept to evaluate the improvements that the fusing of inputs could attain.

The background subtraction-based method, known as Improved Multiple Object Tracking (IMOT) converts blobs of foreground into compact bounding boxes with the consideration of object edges and optical flow. With a filtering step applied to IMOT objects to allow only a certain minimum size for their validity consideration, it is assumed that the inputs provided by the IMOT are more reliable than the ones from the multi-class object detector (MOD) that contains a lot of redundancies. IOU between the IMOT objects and MOD objects are used to determine the validity of an object as an input to be fed into the tracking procedure. A strategy was devised to verify the existence of targets from both sources since the two may complement each other, despite their own sets of individual disadvantages. Since IMOT objects are usually smaller and more compact, it is likely that some of them are fragmented parts of targets. Aside from the comparison between IMOT and MOD objects, comparison of IMOT objects that are in close proximity was also performed and their similarity was evaluated. Color histogram information was taken into consideration when attempting to merge multiple fragmented IMOT objects into one object, resulting in a cleaner set of objects that comes with label information given by the multi-class detector.

Nonetheless, it was observed that the bounding boxes from MOD are still somewhat unreliable as the generated objects have the tendency of containing areas that do not belong to the targets and that are not sufficiently compact. On the other hand, it is useful in cases of partial occlusion, which is highly common due to the presence of pedestrians in urban traffic scenes.

As with any other method of object extraction, the fusing strategy, however, is not perfect for all cases. It is challenging to perform accurate tracking when objects are mistakenly merged as a big object instead of being represented as the distinct objects that they actually are. Occasionally, even the IMOT representation might already contain more than one object of interest due to the nature of the background subtraction approach. A strict threshold was imposed during the merging process to minimize the number of incorrect fusion of objects. During the fusion process, in addition to the distances, the color information is integral in differentiating between the case of combining fragmented parts into one object and the case of having multiple distinct objects interacting in close proximity. Due to the prominent role of color information in such circumstances, we noticed that the excessive inclusion of background areas in the bounding boxes that are not compactly containing the objects of interest impairs the tracking process in the end due to the irrelevant details from the background. Since the proposed method is essentially filtering IMOT objects with MOD objects, it has in effect disregarded the possible objects provided by the MOD that were missed out by the IMOT, which could have been helpful. However, given the redundancies of bounding boxes from our previous experiments, it was not a feasible choice to take that into consideration.

Throughout the tracking process, the Kalman filter produces a prediction from the accumulated history for each object at each time step. Our tracking strategy was fortified by an evaluation on the quality of the prediction to determine if it is indeed appropriate to the circumstances. The predictions are inserted into final trajectories only as needed when the quality of the prediction is sufficiently good or reliable based on the overlap criteria between the prediction result and the previous position in the trajectory history. This allows the flexibility of using the previous result in the history instead of the prediction if it is already known to be unreliable. In the final set of trajectories, the ones with an excessively high number of unreliable and bad predictions are disregarded so that only the trajectories with good quality are outputted as our tracking results.

7.4 Supervised vs unsupervised detections

Finally, given the challenges faced in fusing objects of different sources in Chapter 5, we have decided to take a step back and evaluate objectively the exact effects of using supervised and unsupervised inputs in our tracking framework. Since it was observed that neither bounding boxes that loosely contain the targets (with inclusion of parts of background or other targets) nor overly excessive compact bounding boxes that only capture a part of the object of interest are good enough to overlook their impacts on the tracking process, we experimented on a different detector that gives potentially more accurate inputs for our tracker. In Chapter 6,

a new SOTA object detector, RetinaNet [79] was employed as the source for supervised inputs and it was tested on the UA-Detrac dataset as well. Unsupervised inputs are from IMOT paired with PAWCS (Pixel-based Adaptive Word Consensus Segmenter) [116], a SOTA background subtraction method. Hence, both the supervised and unsupervised inputs are in the form of bounding boxes that enclosed the objects of interest. In addition to the existing features that were re-used from the previous methods (spatial distance, color histogram and object class label), ReID features were included as well to better describe the targets for tracking. Comparison of the tracking performance was also done with other methods from the literature on both datasets with both supervised and unsupervised inputs. From the results, it was interpreted that the supervised inputs generally work very well for datasets with large available training data, such as the UA-Detrac dataset. Certain videos in the dataset worked well with unsupervised inputs too, albeit with slightly lower quantitative results. However, datasets with limited size, such as the Urban Tracker dataset, require a different strategy, seeing that it was not feasible to perform training on the data itself to prevent overfitting, while also subjecting the supervised detector to sufficient variability. It was noted that despite the small size of the dataset, the urban traffic scenarios are inherently very different from one another to accommodate sufficient training. On the other hand, the UA-Detrac dataset provides large scale scenes with similar locations with subtle changes that can facilitate intensive training.

A supervised method like RetinaNet fares better in crowded traffic scenes due to its ability to detect each target individually. An unsupervised method might observe the different road users as a merged entity and thus has difficulty for effective tracking. Nonetheless, in cases where there is an unexpected object that is not learnt prior to detection, an unsupervised method would still be able to perform the extraction whereas a supervised method will not be able to identify it at all.

In short, our proposed tracker has undergone various changes and modifications for improvement over time and is adapted to work with different types of inputs, whether they are originating purely from a supervised method, an unsupervised method or a fusion of the two. We have found some evidence and verified with experiments that the choice of inputs for MOT is not a one-size-fits-all solution, as the type of evaluated dataset determines the most suitable approach to tackle the problem to achieve good tracking results in the end.

CHAPTER 8 CONCLUSION

This chapter concludes the thesis with remarks and insights gained from the research conducted and our experimentation.

8.1 Advancement of knowledge

In this project, we sought to propose a good tracking strategy that addressed the issues and complications encountered in urban traffic scenes. Indeed, assessing the problem has revealed many challenges that resulted from the frequent occurrence of occlusions and interactions among targets in the scenes. To this end, we started to examine and tackle the problem from the very start: the extraction of road users. Through the studies conducted during this research, we have observed and concluded that the quality of the extraction of objects critically determines the level of success in tracking them individually at later stages. Hence, a significant part of the research effort is dedicated on assessing and investigating the manner of obtaining the optimal inputs for tracking purposes.

First, we started by applying a supervised multi-class object detector to supply inputs for our proposed MOT solution, at the same time capitalizing on class label information for the tracking process. More precisely, in order to take into account the variability of object types in the urban traffic scenes, we proposed the use of object class labels and their confidence levels obtained from a deep learning-based object detector as part of the features that describe the objects of interest. We have demonstrated this information to be useful and valuable in improving the performance of the tracker on the evaluated datasets. While the class label information does give satisfactory results in the tracking stage, the quality of bounding boxes as representation of targets provided by the object detector was not up to par with our expectation. Contributions in this part of our work are the incorporation of a modern object detector as well as the use of object class label information in our proposed tracker.

Then, we investigated further more sophisticated manners of combining inputs from different sources, while tweaking our tracker strategy as well for improvements to achieve better tracking results for each of the target in the scene. We acknowledged that there are different pros and cons from different manner of extracting of objects of interest, and we yearned to achieve the best-of-both-world solution by complementing the different approaches. The proposed fusion showed improvements to a certain degree, but there are still occasions where it does not perform optimally. As for our tracking strategy, we continued to use the most

popular data association method, the Hungarian algorithm for matching the objects across frames due to its simplicity and effectiveness. In fact, the vast majority of the newest trackers are still using this proven approach. The prediction strategy that handles occlusion was improved too, with the evaluation of the quality of prediction before its integration into the final trajectory outputs. In short, this second part of our work contributed a fusion strategy for the inputs of a tracker from various sources in addition to a prediction evaluation strategy that improves tracking performance.

Last but not least, we conducted a thorough investigation that compares the effects of supervised and unsupervised inputs, where the experimental comparison involves our own tracker as well as other existing methods on the same datasets. While at first glance, it seemed that the results were contradicting on the two datasets, a deeper analysis further revealed the reasons for such observations. UA-Detrac dataset is larger in size, offering a large number of training frames in rather similar settings as the test frames with slight variations over time. It is ideal for the use of supervised methods to give good tracking results based on the learned inputs. Supervised inputs have an edge over unsupervised inputs in this case as it is not easily affected by the occasional noise caused by the movements of the cameras and environmental perturbations. On the contrary, the Urban Tracker dataset is much limited in size with no training data, but contains scenes that highly varies from one video to another. Our attempts at tracking using supervised detectors was not very successful. Instead, unsupervised inputs for this dataset fared better in our tracking results. In addition, it is important to note that a supervised detector would not be able to detect unexpected road users that are not learned prior training. This allows unsupervised method to be able to capture this occurrence, however rare it might be. Hence, it is concluded that there is no perfect set of inputs that fits all datasets and the flexibility of using different detectors is needed, whether it is supervised or unsupervised, depending on the nature of the data and requirements. Automatic tracking without considering the type of data is not likely to achieve its highest potential in its tracking performance. The last part of our work contributed a comprehensive assessment of the impacts of supervised and unsupervised detection in the context of MOT.

8.2 Limits and recommendations

In the implementation of our proposed MOT solution, the choice of feature weight assigned for data association is obtained in heuristic manner and is dependent on threshold values. In a large scale experiment involving a lot of datasets, it might not be feasible to manually perform the empirical tuning of weights, and thus an automatic approach could be preferable.

Besides, even for a single video, describing all of its objects efficiently might require different sets of weights. Therefore, the proposed method could benefit from a more sophisticated way of combining the features to match the objects of interest across frames. One possible idea for this step is metric learning, where similarity of pairs of objects are trained and learnt before assigning the weights for the feature cost used in the Hungarian algorithm. Background pixels in bounding boxes also affects the quality of the object representation. As a result, it is postulated that a segmented representation that enclose the targets closely might be a more accurate representation of road users for tracking, as illustrated by our earlier attempts of using superpixel representation. It is perhaps where the research direction could be heading towards since we observed lately some recent works perform MOT from segmentation with object masks.

REFERENCES

[1] B. Bose, X. Wang, and E. Grimson, "Multi-class object tracking algorithm that handles fragmentation and grouping," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2007, pp. 1–8.

[2] W. Luo *et al.*, "Multiple object tracking: A literature review," *arXiv preprint arXiv:1409.7618*, 2014.

[3] S. Lyu *et al.*, "Ua-detrac 2018: Report of avss2018 & iwt4s challenge on advanced traffic monitoring," in *2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. IEEE, 2018, pp. 1–6.

[4] T. Brox *et al.*, "High accuracy optical flow estimation based on a theory for warping," in *European conference on computer vision*. Springer, 2004, pp. 25–36.

[5] United Nations, "World population prospects: The 2015 revision," *United Nations Economic Social Affairs*, vol. 33, no. 2, pp. 1–66, 2015.

[6] L. Chapman, "Transport and climate change: a review," *Journal of transport geography*, vol. 15, no. 5, pp. 354–367, 2007.

[7] M. Barth and K. Boriboonsomsin, "Real-world carbon dioxide impacts of traffic congestion," *Transportation Research Record: Journal of the Transportation Research Board*, no. 2058, pp. 163–171, 2008.

[8] A. A. Cruz, *Global surveillance, prevention and control of chronic respiratory diseases: a comprehensive approach*. World Health Organization, 2007.

[9] D. A. Hennessy and D. L. Wiesenthal, "The relationship between traffic congestion, driver stress and direct versus indirect coping behaviours," *Ergonomics*, vol. 40, no. 3, pp. 348–361, 1997.

[10] ——, "Traffic congestion, driver stress, and driver aggression," *Aggressive Behavior: Official Journal of the International Society for Research on Aggression*, vol. 25, no. 6, pp. 409–423, 1999.

[11] L. Wen *et al.*, "UA-DETRAC: A new benchmark and protocol for multi-object detection and tracking," *Computer Vision and Image Understanding*, 2020.

[12] A. Andriyenko, S. Roth, and K. Schindler, "An analytical formulation of global occlusion reasoning for multi-target tracking," in *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*. IEEE, 2011, pp. 1839–1846.

[13] H. Possegger *et al.*, "Occlusion geodesics for online multi-object tracking," in *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1306–1313.

[14] S. Chen, A. Fern, and S. Todorovic, "Multi-object tracking via constrained sequential labeling," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1130–1137.

[15] K. Fragkiadaki *et al.*, "Two-granularity tracking: Mediating trajectory and detection graphs for tracking under occlusions," in *European Conference on Computer Vision*. Springer, 2012, pp. 552–565.

[16] P. F. Gabriel *et al.*, "The state of the art in multiple object tracking under occlusion in video sequences," in *Advanced Concepts for Intelligent Vision Systems*, 2003, pp. 166–173.

[17] J.-W. Hsieh *et al.*, "Automatic traffic surveillance system for vehicle tracking and classification," *IEEE Transactions on Intelligent Transportation Systems*, vol. 7, no. 2, pp. 175–187, 2006.

[18] J.-C. Lai, S.-S. Huang, and C.-C. Tseng, "Image-based vehicle tracking and classification on the highway," in *The 2010 International Conference on Green Circuits and Systems*. IEEE, 2010, pp. 666–670.

[19] S.-H. Bae and K.-J. Yoon, "Robust online multi-object tracking based on tracklet confidence and online discriminative appearance learning," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 1218–1225.

[20] J.-P. Jodoin, G.-A. Bilodeau, and N. Saunier, "Tracking all road users at multimodal urban traffic intersections," *IEEE transactions on intelligent transportation systems*, vol. 17, no. 11, pp. 3241–3251, 2016.

[21] S.-H. Bae and K.-J. Yoon, "Confidence-based data association and discriminative deep appearance learning for robust online multi-object tracking," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 3, pp. 595–610, 2017.

[22] A. Bewley *et al.*, "Simple online and realtime tracking," in *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2016, pp. 3464–3468.

[23] Q. Chu *et al.*, "Online multi-object tracking using cnn-based single object tracker with spatial-temporal attention mechanism," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 4836–4845.

[24] P. Chu and H. Ling, "Famnet: Joint learning of feature, affinity and multi-dimensional assignment for online multiple object tracking," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 6172–6181.

[25] Z. Lu *et al.*, "Retinatrack: Online single stage joint detection and tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 14 668–14 678.

[26] S. Tang *et al.*, "Subgraph decomposition for multi-target tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5033–5041.

[27] ——, "Multi-person tracking by multicut and deep matching," in *European Conference on Computer Vision*. Springer, 2016, pp. 100–111.

[28] ——, "Multiple people tracking by lifted multicut and person re-identification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3539–3548.

[29] C. Kim *et al.*, "Multiple hypothesis tracking revisited," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4696–4704.

[30] W. Choi, "Near-online multi-target tracking with aggregated local flow descriptor," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 3029–3037.

[31] L. Wen *et al.*, "Learning non-uniform hypergraph for multi-object tracking," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 8981–8988.

[32] N. Saunier and T. Sayed, "A feature-based tracking algorithm for vehicles in intersections," in *The 3rd Canadian Conference on Computer and Robot Vision (CRV'06)*. IEEE, 2006, pp. 59–59.

[33] S. Birchfield, "An implementation of the kanade-lucas-tomasi feature tracker," 1998.

[34] Y. Yang and G.-A. Bilodeau, "Multiple object tracking with kernelized correlation filters in urban mixed traffic," in *2017 14th Conference on Computer and Robot Vision (CRV)*. IEEE, 2017, pp. 209–216.

[35] D.-A. Beaupré, G.-A. Bilodeau, and N. Saunier, "Improving multiple object tracking with optical flow and edge preprocessing," *arXiv preprint arXiv:1801.09646*, 2018.

[36] J. Li, X. Gao, and T. Jiang, "Graph networks for multiple object tracking," in *The IEEE Winter Conference on Applications of Computer Vision*, 2020, pp. 719–728.

[37] N. F. Gonzalez, A. Ospina, and P. Calvez, "Smat: Smart multiple affinity metrics for multiple object tracking," in *International Conference on Image Analysis and Recognition*. Springer, 2020, pp. 48–62.

[38] S. Ren *et al.*, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.

[39] J. Ren *et al.*, "Accurate single stage detector using recurrent rolling convolution," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 5420–5428.

[40] F. Yang, W. Choi, and Y. Lin, "Exploit all the layers: Fast and accurate cnn object detector with scale dependent pooling and cascaded rejection classifiers," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2129–2137.

[41] M. Babaee, A. Athar, and G. Rigoll, "Multiple people tracking using hierarchical deep tracklet re-identification," *arXiv preprint arXiv:1811.04091*, 2018.

[42] K. He *et al.*, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[43] R. Henschel, Y. Zou, and B. Rosenhahn, "Multiple people tracking using body and joint detections," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2019, pp. 0–0.

[44] M. Andriluka *et al.*, "Posetrack: A benchmark for human pose estimation and tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5167–5176.

[45] D. Riahi and G.-A. Bilodeau, "Multiple feature fusion in the dempster-shafer framework for multi-object tracking," in *2014 Canadian Conference on Computer and Robot Vision*. IEEE, 2014, pp. 313–320.

[46] F. Yu *et al.*, "Poi: Multiple object tracking with high performance detection and appearance feature," in *European Conference on Computer Vision*. Springer, 2016, pp. 36–42.

[47] F. Solera, S. Calderara, and R. Cucchiara, "Learning to divide and conquer for online multi-target tracking," in *proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 4373–4381.

[48] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *2017 IEEE international conference on image processing (ICIP)*. IEEE, 2017, pp. 3645–3649.

[49] X. Zhou *et al.*, "Online multi-object tracking with structural invariance constraint." in *BMVC*, 2018, p. 203.

[50] R. Sanchez-Matilla, F. Poiesi, and A. Cavallaro, "Online multi-target tracking with strong and weak detections," in *European Conference on Computer Vision*. Springer, 2016, pp. 84–99.

[51] E. Ristani and C. Tomasi, "Features for multi-target multi-camera tracking and re-identification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 6036–6046.

[52] E. Bochinski, V. Eiselein, and T. Sikora, "High-speed tracking-by-detection without using image information," in *International Workshop on Traffic and Street Surveillance for Safety and Security at IEEE AVSS 2017*, Lecce, Italy, Aug. 2017. [Online]. Available: http://elvera.nue.tu-berlin.de/files/1517Bochinski2017.pdf

[53] E. Bochinski, T. Senst, and T. Sikora, "Extending iou based multi-object tracking by visual information," in *IEEE International Conference on Advanced Video and Signals-based Surveillance*, Auckland, New Zealand, Nov. 2018, pp. 441–446. [Online]. Available: http://elvera.nue.tu-berlin.de/files/1547Bochinski2018.pdf

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

[55] P. Chu *et al.*, "Online multi-object tracking with instance-aware tracker and dynamic model refreshment," in *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2019, pp. 161–170.

[56] Y. Xu *et al.*, "A causal and-or graph model for visibility fluent reasoning in tracking interacting objects," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2178–2187.

[57] I. Newton and J. Colson, *The Method of Fluxions and Infinite Series; with Its Application to the Geometry of Curve-lines... Translated from the Author's Latin Original Not Yet Made Publick. To which is Subjoin'd a Perpetual Comment Upon the Whole Work... by J. Colson*, 1736.

[58] H. Y. Ju *et al.*, "Online multi-object tracking via structural constraint event aggregation," in *Proceedings of the IEEE Conference on computer vision and pattern recognition*, 2016, pp. 1392–1400.

[59] H. Kieritz, W. Hubner, and M. Arens, "Joint detection and online multi-object tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 1459–1467.

[60] A. Milan *et al.*, "Online multi-target tracking using recurrent neural networks," in *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[61] J. Son *et al.*, "Multi-object tracking with quadruplet convolutional neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 5620–5629.

[62] L. Ren *et al.*, "Collaborative deep reinforcement learning for multi-object tracking," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 586–602.

[63] Z. Zhang *et al.*, "Integrated object detection and tracking with tracklet-conditioned detection," *arXiv preprint arXiv:1811.11167*, 2018.

[64] S. Sun *et al.*, "Deep affinity network for multiple object tracking," *IEEE transactions on pattern analysis and machine intelligence*, 2019.

[65] J. Zhu *et al.*, "Online multi-object tracking with dual matching attention networks," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 366–382.

[66] Y. Xiang, A. Alahi, and S. Savarese, "Learning to track: Online multi-object tracking by decision making," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4705–4713.

[67] A. Sadeghian, A. Alahi, and S. Savarese, "Tracking the untrackable: Learning to track multiple cues with long-term dependencies," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 300–311.

[68] C. Ma *et al.*, "Trajectory factory: Tracklet cleaving and re-connection by deep siamese bi-gru for multiple object tracking," in *2018 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2018, pp. 1–6.

[69] S. Schulter *et al.*, "Deep network flow for multi-object tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6951–6960.

[70] K. Fang *et al.*, "Recurrent autoregressive networks for online multi-object tracking," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2018, pp. 466–475.

[71] X. Shi *et al.*, "Rank-1 tensor approximation for high-order association in multi-target tracking," *International Journal of Computer Vision*, vol. 127, no. 8, pp. 1063–1083, 2019.

[72] P. Bergmann, T. Meinhardt, and L. Leal-Taixe, "Tracking without bells and whistles," in *Proceedings of the IEEE international conference on computer vision*, 2019, pp. 941–951.

[73] X. Zhou, V. Koltun, and P. Krähenbühl, "Tracking objects as points," *ECCV*, 2020.

[74] Z. Wang *et al.*, "Towards real-time multi-object tracking," *ECCV*, 2019.

[75] G. Wang *et al.*, "Exploit the connectivity: Multi-object tracking with trackletnet," in *Proceedings of the 27th ACM International Conference on Multimedia*, 2019, pp. 482–490.

[76] S. Sun *et al.*, "Simultaneous detection and tracking with motion modelling for multiple object tracking," *ECCV*, 2020.

[77] W. Liu *et al.*, "Ssd: Single shot multibox detector," in *European conference on computer vision*. Springer, 2016, pp. 21–37.

[78] J. Peng *et al.*, "Chained-tracker: Chaining paired attentive regression results for end-to-end joint multiple-object detection and tracking," *ECCV*, 2020.

[79] T.-Y. Lin *et al.*, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

[80] G. Brasó and L. Leal-Taixé, "Learning a neural solver for multiple object tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 6247–6257.

[81] S. Leutenegger, M. Chli, and R. Y. Siegwart, "Brisk: Binary robust invariant scalable keypoints," in *2011 International conference on computer vision*. Ieee, 2011, pp. 2548–2555.

[82] A. Alahi, R. Ortiz, and P. Vandergheynst, "Freak: Fast retina keypoint," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. Ieee, 2012, pp. 510–517.

[83] J. Xu *et al.*, "Spatial-temporal relation networks for multi-object tracking," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 3988–3998.

[84] C. Kim, F. Li, and J. M. Rehg, "Multi-object tracking with neural gating using bilinear lstm," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 200–215.

[85] C. Li *et al.*, "Tracknet: Simultaneous object detection and tracking and its application in traffic video analysis," *arXiv preprint arXiv:1902.01466*, 2019.

[86] B. D. Lucas, T. Kanade *et al.*, "An iterative image registration technique with an application to stereo vision," in *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*. Vancouver, British Columbia, 1981.

[87] Y. Xiang, R. Mottaghi, and S. Savarese, "Beyond pascal: A benchmark for 3d object detection in the wild," in *IEEE winter conference on applications of computer vision*. IEEE, 2014, pp. 75–82.

[88] O. Styles, V. Sanchez, and T. Guha, "Multiple object forecasting: Predicting future object locations in diverse environments," in *The IEEE Winter Conference on Applications of Computer Vision*, 2020, pp. 690–699.

[89] T. Fernando *et al.*, "Tracking by prediction: A deep generative model for mutli-person localisation and tracking," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2018, pp. 1122–1132.

[90] K. Hara, H. Kataoka, and Y. Satoh, "Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet?" in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2018, pp. 6546–6555.

[91] K. Bernardin and R. Stiefelhagen, "Evaluating multiple object tracking performance: the clear mot metrics," *EURASIP Journal on Image and Video Processing*, vol. 2008, pp. 1–10, 2008.

[92] J. Dai *et al.*, "R-fcn: Object detection via region-based fully convolutional networks," in *Advances in neural information processing systems*, 2016, pp. 379–387.

[93] J. Redmon *et al.*, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[94] L. M. Fuentes and S. A. Velastin, "People tracking in surveillance applications," *Image and Vision Computing*, vol. 24, no. 11, pp. 1165–1171, 2006.

[95] A. Torabi and G.-A. Bilodeau, "A multiple hypothesis tracking method with fragmentation handling," in *2009 Canadian Conference on Computer and Robot Vision*. IEEE, 2009, pp. 8–15.

[96] G. Jun, J. Aggarwal, and M. Gokmen, "Tracking and segmentation of highway vehicles in cluttered and crowded scenes," in *2008 IEEE Workshop on Applications of Computer Vision*. IEEE, 2008, pp. 1–6.

[97] Z. Kim, "Real time object tracking based on dynamic feature grouping with background subtraction," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2008, pp. 1–8.

[98] J. C. Mendes, A. G. C. Bianchi, and Á. R. P. Júnior, "Vehicle tracking and origin-destination counting system for urban environment." in *VISAPP (3)*, 2015, pp. 600–607.

[99] J. Shi *et al.*, "Good features to track," in *1994 Proceedings of IEEE conference on computer vision and pattern recognition*. IEEE, 1994, pp. 593–600.

[100] D. Beymer *et al.*, "A real-time computer vision system for measuring traffic parameters," in *Proceedings of IEEE computer society conference on computer vision and pattern recognition*. IEEE, 1997, pp. 495–501.

[101] B. Coifman *et al.*, "A real-time computer vision system for vehicle tracking and traffic surveillance," *Transportation Research Part C: Emerging Technologies*, vol. 6, no. 4, pp. 271–288, 1998.

[102] S. Aslani and H. Mahdavi-Nasab, "Optical flow based moving object detection and tracking for traffic surveillance," *International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering*, vol. 7, no. 9, pp. 1252–1256, 2013.

[103] Z. Luo *et al.*, "Mio-tcd: A new benchmark dataset for vehicle classification and localization," *IEEE Transactions on Image Processing*, vol. 27, no. 10, pp. 5129–5141, 2018.

[104] T. Malisiewicz, A. Gupta, and A. A. Efros, "Ensemble of exemplar-svms for object detection and beyond," in *2011 International conference on computer vision*. IEEE, 2011, pp. 89–96.

[105] P. F. Felzenszwalb *et al.*, "Object detection with discriminatively trained part-based models," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 9, pp. 1627–1645, 2009.

[106] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, 1960.

[107] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval research logistics quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.

[108] R. Girshick *et al.*, "Region-based convolutional networks for accurate object detection and segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 1, pp. 142–158, 2016.

[109] S. Hamid Rezatofighi *et al.*, "Joint probabilistic data association revisited," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 3047–3055.

[110] Y. Bar-Shalom, F. Daum, and J. Huang, "The probabilistic data association filter," *IEEE Control Systems Magazine*, vol. 29, no. 6, pp. 82–100, 2009.

[111] H. Pirsiavash, D. Ramanan, and C. C. Fowlkes, "Globally-optimal greedy algorithms for tracking a variable number of objects," in *CVPR 2011*. IEEE, 2011, pp. 1201–1208.

[112] J. F. Henriques *et al.*, "High-speed tracking with kernelized correlation filters," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 3, pp. 583–596, 2014.

[113] A. Milan *et al.*, "Mot16: A benchmark for multi-object tracking," *arXiv preprint arXiv:1603.00831*, 2016.

[114] L. Wang *et al.*, "Evolving boxes for fast vehicle detection," in *2017 IEEE international conference on multimedia and Expo (ICME)*. IEEE, 2017, pp. 1135–1140.

[115] O. Barnich and M. Van Droogenbroeck, "Vibe: A universal background subtraction algorithm for video sequences," *IEEE Transactions on Image processing*, vol. 20, no. 6, pp. 1709–1724, 2010.

[116] P.-L. St-Charles, G.-A. Bilodeau, and R. Bergevin, "Universal background subtraction using word consensus models," *IEEE Transactions on Image Processing*, vol. 25, no. 10, pp. 4768–4781, 2016.

[117] M. D. Breitenstein *et al.*, "Online multiperson tracking-by-detection from a single, uncalibrated camera," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 9, pp. 1820–1833, 2010.

[118] K. Zhou *et al.*, "Omni-scale feature learning for person re-identification," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 3702–3712.

[119] L. Wei *et al.*, "Person transfer gan to bridge domain gap for person re-identification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 79–88.

[120] R. Girshick *et al.*, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[121] R. Achanta *et al.*, "Slic superpixels compared to state-of-the-art superpixel methods," *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 11, pp. 2274–2282, 2012.

[122] M. Van den Bergh *et al.*, "Seeds: Superpixels extracted via energy-driven sampling," in *European conference on computer vision*. Springer, 2012, pp. 13–26.