

Dynamic 3D Representations and Robust Evaluation for Visual Tracking

Von der Fakultät für Mathematik, Informatik und Naturwissenschaften der
RWTH Aachen University zur Erlangung des akademischen Grades
eines Doktors der Naturwissenschaften genehmigte Dissertation

vorgelegt von

Jonathon Luiten, M. Sc.

aus Whangarei, New Zealand

Berichter: Universitätsprofessor Dr. sc. techn. Bastian Leibe
Professor Deva Ramanan

Tag der mündlichen Prüfung: 6. March 2024

Diese Dissertation ist auf den Internetseiten der Universitätsbibliothek online verfügbar.

Abstract

Visual tracking is a core task within computer vision, one that involves understanding the motion and persistence of the dynamic world when observed in video. Building performant tracking algorithms is critical for many applications such as robotics; self-driving vehicles; virtual and augmented reality; scene-analysis for sporting, retail and construction scenarios; and content creation and editing. While other areas of computer vision such as recognition and detection have recently reached outstanding performance due to deep learning and extremely large training datasets, tracking has remained an incredibly difficult task where such approaches have not been able to achieve similar success. We argue that tracking is inherently different from these tasks and that simply scaling up compute and data is not going to be enough. In this thesis we develop what we believe to be the missing piece holding tracking back from similar success: the use of dynamic 3D representations that can be used to model the underlying scene. Furthermore, we find that the second critical thing to address in the field of visual tracking is the lack of adequate evaluation metrics and benchmark settings. We address these limitations by introducing novel metrics and benchmarks, which are crucial for measuring the performance of algorithms and guiding the field toward making meaningful progress.

The first half of this thesis deals with approaches to lift representations for tracking to 3D, both at the level of whole objects (MOTSFusion) and at the level of infinitesimal 3D scene elements (Dynamic 3D Gaussians). Traditionally, tracking involves finding correspondences between static 2D representations in each timestep, such as pixels or bounding-boxes. Instead, we represent the world as a set of dynamic 3D representations, that move around over time in order to consistently represent the same physical location in space as it moves. We reformulate tracking from a correspondence estimation problem, to an analysis-by-synthesis problem of fitting an underlying dynamic 3D model, whose motion explains changes in image content across timesteps. By using

3D representations we can better model appearance changes due to the 3D motion of the scene and the motion of the camera through the world, while also making use of intuitive physics knowledge about how objects move through the 3D world. This enables us to both obtain better tracking results, while also resulting in consistent dynamic 3D representations that are directly useful for many downstream tasks.

The second half of this thesis deals with building robust metrics and benchmarks for evaluating the performance of visual tracking algorithms. For the task of Multi-Object Tracking (MOT), previous evaluation metrics have been sorely lacking, focusing only on particular aspects of tracking performance (e.g. detection or association), but not being able holistically measure improvements in tracking performance. Furthermore, tracking evaluation has been limited to setting where only a small number of fixed object classes were evaluated. We address both of these evaluation limitations by proposing the HOTA Metrics for evaluating tracking performance in a fair and holistic way, and introducing the task of Open-World Tracking for extending tracking evaluation to an open-world setting where a potentially unlimited set of object classes need to be tracked, even if they were not previously seen during training. Together, these mark a step-change in how tracking methods are evaluated and benchmarked, and allow the tracking community to make meaningful progress towards more performant and useful tracking algorithms.

Overall, by developing both dynamic 3D representations for tracking and a novel set of evaluation metrics and benchmarks, this thesis provides a number of crucial missing pieces that are needed to move towards truly useful and performant tracking algorithms, and thus toward the success of the multitude of applications for which tracking is a core component.

Zusammenfassung

Die visuelle Verfolgung ist eine zentrale Aufgabe im Bereich der Computer Vision, bei der es darum geht, die Bewegung und die Beständigkeit der dynamischen Welt in einem Video zu verstehen. Die Entwicklung leistungsfähiger Verfolgungsalgorithmen ist für viele Anwendungen von entscheidender Bedeutung, z. B. in der Robotik, bei selbstfahrenden Fahrzeugen, in der virtuellen und erweiterten Realität, bei der Szenenanalyse für Sport-, Einzelhandels- und Bauszenarien sowie bei der Erstellung und Bearbeitung von Inhalten. Während andere Bereiche der Computer Vision, wie z.B. die Erkennung und Detektion, in letzter Zeit dank Deep Learning und extrem großen Trainingsdatensätzen herausragende Leistungen erreicht haben, ist das Tracking eine unglaublich schwierige Aufgabe geblieben, bei der solche Ansätze nicht in der Lage waren, ähnliche Erfolge zu erzielen. Wir argumentieren, dass sich die Verfolgung von Natur aus von diesen Aufgaben unterscheidet und dass eine einfache Skalierung von Rechenleistung und Daten nicht ausreicht. In dieser Arbeit entwickeln wir, was unserer Meinung nach das fehlende Element ist, das dem Tracking einen ähnlichen Erfolg verwehrt: die Verwendung von dynamischen 3D-Darstellungen, die zur Modellierung der zugrunde liegenden Szene verwendet werden können. Darüber hinaus stellen wir fest, dass das zweite Hindernis im Bereich des visuellen Trackings das Fehlen geeigneter Evaluierungsmetriken und Benchmarking-Einstellungen ist. Wir setzen uns mit diesen Einschränkungen auseinander, indem wir neuartige Metriken und Benchmarks einführen, die für die Messung der Leistung von Algorithmen entscheidend sind und das Feld zu sinnvollen Fortschritten führen.

Die erste Hälfte dieser Arbeit befasst sich mit Ansätzen zur Übertragung von Darstellungen für die Verfolgung auf 3D, sowohl auf der Ebene ganzer Objekte (MOTSFusion) als auch auf der Ebene infinitesimaler 3D-Szenenelemente (Dynamic 3D Gaussians). Traditionell werden beim Tracking in jedem Zeitschritt Korrespondenzen zwischen

statischen 2D-Darstellungen wie Pixeln oder Bounding-Boxen gefunden. Stattdessen stellen wir die Welt als eine Menge dynamischer 3D-Darstellungen dar, die sich im Laufe der Zeit bewegen, um konsistent denselben physischen Ort im Raum zu repräsentieren, während er sich bewegt. Wir formulieren das Tracking von einem Korrespondenzschätzungsproblem zu einem Analyse-durch-Synthese-Problem um, bei dem es um die Anpassung eines zugrunde liegenden dynamischen 3D-Modells geht, dessen Bewegung Änderungen des Bildinhalts über Zeitschritte hinweg erklärt. Durch die Verwendung von 3D-Darstellungen können wir Erscheinungsänderungen aufgrund der 3D-Bewegung der Szene und der Bewegung der Kamera durch die Welt besser modellieren und gleichzeitig intuitives physikalisches Wissen darüber nutzen, wie sich Objekte durch die 3D-Welt bewegen. Auf diese Weise können wir sowohl bessere Verfolgungsergebnisse erzielen als auch konsistente dynamische 3D-Darstellungen, die für viele nachgelagerte Aufgaben direkt nützlich sind.

Die zweite Hälfte dieser Arbeit befasst sich mit der Entwicklung robuster Metriken und Benchmarks zur Bewertung der Leistung visueller Tracking-Algorithmen. Für die Aufgabe des Multi-Objekt-Trackings (MOT) fehlten bisher Evaluierungsmetriken, die sich nur auf bestimmte Aspekte der Tracking-Leistung (z.B. Erkennung oder Assoziation) konzentrieren, aber nicht in der Lage sind, Verbesserungen der Tracking-Leistung ganzheitlich zu messen. Darüber hinaus war die Bewertung der Verfolgungsleistung auf Situationen beschränkt, in denen nur eine kleine Anzahl von festen Objektklassen bewertet wurde. Die HOTA-Metriken ermöglichen eine faire und ganzheitliche Bewertung der Tracking-Leistung, und die Open-World-Tracking-Aufgabe erweitert die Tracking-Evaluierung auf eine offene Welt, in der eine potenziell unbegrenzte Anzahl von Objektklassen verfolgt werden muss, selbst wenn diese zuvor beim Training nicht gesehen wurden. Zusammen markieren diese Maßnahmen eine Veränderung in der Art und Weise, wie Tracking-Methoden bewertet und verglichen werden, und ermöglichen es der Tracking-Gemeinschaft, sinnvolle Fortschritte auf dem Weg zu leistungsfähigeren und nützlicheren Tracking-Algorithmen zu machen.

Insgesamt liefert diese Arbeit durch die Entwicklung von dynamischen 3D Darstellungen für das Tracking und einer neuartigen Reihe von Evaluierungsmetriken und Benchmarks eine Reihe von entscheidenden fehlenden Teilen, die benötigt werden, um wirklich nützliche und leistungsfähige Tracking-Algorithmen zu entwickeln und damit den Erfolg der Vielzahl von Anwendungen, für die das Tracking eine Kernkomponente ist, zu gewährleisten.

Acknowledgments

Every PhD is a journey, and any journey would not be complete without companions to share it with. What a journey it has been, and what great companions I have had along the way. My five year PhD journey has been roughly split into five separate one year journeys across space and time: the 1st year in Aachen, the 2nd in Oxford, the 3rd remote during the pandemic in New Zealand, Poland and Germany, the 4th at Carnegie Mellon in Pittsburgh, and the 5th remote as a digital nomad, spending most of my time between Boston, San Francisco, Pittsburgh, New Zealand and Germany.

I would like to begin by thanking **my partner**, Joanna Materzynska, who I actually met at the CVPR computer vision conference in 2018, a few months before starting this PhD journey (our first interaction being me taking her photo together with Elon Musk). For the whole five years of this thesis she has been by my side supporting me the whole way. Many of the opportunities and collaborations I have had during this thesis have been because of me following her around the world, and meeting various people through her. As she is also a fellow computer vision researcher she has not only been there to support me in my personal life, but also in my research life throughout this PhD, doing countless last-minute reads and edits of my various papers, and constantly discussing interesting research ideas. Thank you from the bottom of my heart.

I would like to thank all my **advisors and mentors** throughout this PhD. First and foremost my Professor at the RWTH Aachen, Prof. Bastian Leibe, who has always supported me in chasing my research dreams and always given me the freedom, resources, kindness and advice that has enabled me to do so. Secondly, my advisor during the last two years of my PhD at Carnegie Mellon University, Prof. Deva Ramaman, who was always quick to understand what I was doing, what I was trying to do, and gave great advice weekly on how to best move toward my goals. Thirdly,

my advisor during the year I spent at the University of Oxford, Prof. Phil Torr, who has been a role-model and supporter both in life and in research. I would also like to thank Prof. Laura Leal-Taixé for her mentorship on three particular projects (HOTA, Open-World Tracking and FutureDet), and Prof. Andreas Geiger for his mentorship on two projects (MOTS and HOTA). These projects would not have been the same without their support and advice. Finally I would like to thank Paul Voigtlaender and Aljoša Ošep who, as more senior PhD students in Aachen, took me under their wing at the beginning of my PhD and taught me most of what I know about computer vision, deep learning and research in general.

Next I would like to thank all of the **students** that I have been fortunate to supervise. During this time I have officially supervised seven Masters students at the RWTH, as well as one summer intern at CMU, including Mark Weber, Tobias Fischer, Idil Esen Zulfikar, Valentin Steiner, Thomas Jantos, Sourabh Swain, Arne Hoffhues, and Jana Zeller. Each of them has likely taught me more than I have taught them and I am forever thankful for the opportunity to work with them.

I would like to thank my other paper author **collaborators** over the years. In roughly chronological order in the collaborations: Stefan Breuers, Michael Krause, Berin Balachandar Gnana Sekar, Patrick Dendorfer, Neehar Peri, Martin Li, Ali Athar, Alexander Hermans, Yang Liu, Achal Dave, Tarasha Khunera, Georgios Kopanas, Nikhil Keetha, Jay Karhade, Krishna Murthy Jatavallabhula, Gengshan Yang and Sebastian Scherer. I would like to highlight my acknowledgement of Ali Athar, who toward the end of my PhD I worked incredibly closely with on three projects (HODOR, TarVIS, BURST), and who swooped in at the right time to save the BURST project when I got too busy with other things. I would also like to highlight Achal Dave (with whom I collaborated with on Open-World Tracking and BURST), who opened my eyes to many things previously missed and introduced me to CMU and Deva. Finally I would like to highlight Georgios Kopanas whose assistance on my final Dynamic 3D Gaussians project proved invaluable.

I would like to thank the many **other researchers** that I did not get a chance to formerly collaborate with, but through discussion have greatly influenced my ideas about research and computer vision. Below in alphabetical order by location, and with apologies to those that I almost certainly forgot:

Bastian's group in Aachen: Alexey Nekrasov, Christian Schmidt, Ciwan Ceylan, Dan Jia, Ekin Celikkan, Francis Engelmann, George Lydakis, István Sáránci, Jens Piekenbrinck, Jonas Schult, Karim Abou Zeid, Lucas Beyer, Markus Knoche, Sabarinath Mahadevan, Stephanie Käs, Theodora Kontogianni, Umer Rafi and Wolfgang Mehner.

Various people from my time in Oxford: Alexei (Alyosha) Efros, Allan Jabri, Andrew Gambardella, Bradley Gram-Hansen, Csaba Botos, Elliot Wu, Fabian Fuchs, Jack

Valmadre, João Henriques, Jishnu Mukhoti, Naeemullah Khan, Weidi Xie and Yuge (Jimmy) Shi.

Deva's Group at CMU: Aayush Bansal, Abhinav Agarwalla, Andrew Saba, Anirudh Chakravarthy, Arun Balajee Vasudevan, Chittesh Thavamani, Chonghyuk (Andrew) Song, Gautam Gare, Haithem Turki, Jason Zhang, Jeet Kanjani, Jeff Tan, Kangle Deng, Nadine Chang, Nathaniel Chodosh, Peiyun Hu, Shu Kong, Shubham Gupta, Xindi Wu and Zhiqiu Lin.

And others at CMU: Adam Harley, Adithya Pediredla, Akash Sharma, Dinesh Reddy, Dorian Chan, Fernando De La Torre, Gaurav Parmar, Kris Kitani, Laszlo Jeni, Leonid Keselman, Mark Sheinin, Mihir Prabhudesai, Raunaq Bhirangi, Rawal Khirodkar, Sheng-Yu Wang, Shubham Tulsiani, Swapnil Pande and Yehonathan Litman.

Of this list of research influences, I would like to highlight Jonas Schult, Mark Sheinin, Dinesh Reddy, Gaurav Parmar and Leonid Keselman as having a particularly large influence on my research taste and direction and my thoughts about the universe in general.

Last, but almost certainly not least, I would like to thank my **family and closest non-research friends**. To my parents Bernard and Carol Luiten, I obviously owe you everything, and I cannot imagine having better parents. Sorry that I have mostly been so far from home this whole time. To my two younger brothers Ryan and Dion Luiten, being your older sibling has been a joy, especially as we have slowly all grown up to become adults together. My various flatmates over the years, particularly Jonathan Schieren, Leona Rodenkirchen and Dan Molenhouse, have been a source of constant joy and companionship in life, lighting up the days and nights of this PhD journey. Finally my closest friends from my undergrad before I started this PhD who have stuck with and supported me throughout: Michael Kapteyn, Oliver Lin, Arya Malek, Sam Caldwell, Jack McIvor, Jung Shaan Lee, Toby Jackson and Ola Shahin.

Contents

1	Introduction	1
1.1	Motivation	1
1.2	Contributions	2
1.3	Background	3
1.4	Outline	4
1.5	Additional Works	5
2	MOTSFusion: Track to Reconstruct and Reconstruct to Track	9
2.1	Introduction	10
2.2	Related Work	12
2.3	Our Approach	14
2.4	Experiments	19
2.5	Discussion	25
3	Dynamic 3D Gaussians: Tracking by Persistent Dynamic View Synthesis	27
3.1	Introduction	27
3.2	Related Work	30
3.3	Method	32
3.4	Experiments	39
3.5	Discussion	45
4	HOTA: A Higher Order Metric for Evaluating Multi-Object Tracking	47
4.1	Introduction	47
4.2	Related Work	48
4.3	Preliminaries	52
4.4	Overview of Previous Metrics	55
4.5	The HOTA Evaluation Metric	58

4.6	Decomposing HOTA into Different Error Types.	62
4.7	Analysing the Design Space of HOTA	66
4.8	HOTA Extensions	72
4.9	Analysing Previous Evaluation Metrics.	76
4.10	Evaluating Trackers with HOTA on MOTChallenge	90
4.11	Human Visual Assessment Study	95
4.12	Discussion	99
5	Opening up Open-World Tracking	101
5.1	Introduction	101
5.2	Related Work	104
5.3	Opening up Open World Tracking	106
5.4	TAO-OW Benchmark	110
5.5	Designing Open-World Trackers	113
5.6	Evaluation	121
5.7	Discussion	122
6	Conclusion	125
6.1	Summary and Contributions	125
6.2	Perspectives	125
	Bibliography	127

Introduction

1.1 Motivation

Computer Vision. Computer Vision is the process of using a computer to understand the world by analysing visual sensor data i.e., images or videos. Computer vision is critical for many applications such as robotics and self-driving vehicles, which need to use computer vision to understand their surrounding world by analysing images that they capture from onboard cameras in order to use this information to plan their actions through space, to avoid obstacles and to complete their prescribed tasks. Computer vision is also used to build virtual worlds from the real world, which can then be used in virtual or augmented reality, in movie editing or creation or for the purpose of analysing scenarios such as sporting events, building sites or retail locations. Computer Vision is a very challenging goal to solve. It requires analysing highly complex and high dimensional visual signals and from these somehow building up an understanding of what and where everything is in the world. While extremely difficult, the rewards for making progress in computer vision are great, with many so many downstream tasks and applications whose improvement would greatly benefit the world.

Visual Tracking. Computer Vision consists of many different sub-tasks. For example one task might be to label whether an image contains a cat. Another might be to estimate the depth of each pixel in an image. In this thesis we are focused on the task of Visual Tracking. Visual Tracking is the task of determining the motion of the scene over time. This can manifest itself in different formulations at different levels of granularity. One example is, for each pixel in one frame of a video, determine where the content of that pixel is in every other frame of a video. Another example is finding every car in a video, while consistently determining where each car is in every video timestep. Tracking is one of the core fundamental tasks of computer vision, and becomes useful and necessary whenever dealing with dynamic data where the world is evolving over

time (e.g. video). And yet, compared to other tasks within computer vision such as recognition or detection, current approaches to tracking are lacking in accuracy and performance and as such limiting the usefulness of many downstream applications. In this thesis we seek to make up these limitations in previous tracking approaches and move tracking toward being more useful and performant.

Dynamic 3D Reconstruction and Representation. 3D Reconstruction is the task of building a 3D model of the geometry of the world from a set of 2D images. This has most commonly been approached for static scenes, where a 3D model can be useful many downstream tasks such as prediction, planning, re-orientation, collision detection, and 3D printing to name a few. Dynamic 3D reconstruction is the task doing the 3D reconstruction task for dynamic scenes, where the underlying geometry changes over time. This task is inherently tied to tracking, in that tracking gives the correspondences needed to perform such a reconstruction, and that the dynamic reconstruction itself is a solution to the tracking problem, detailing how each part of the scene moves over time. A core question in both dynamic and static reconstruction is which underlying representation to use to represent the (possibly moving) geometry. This is a question we investigate in this thesis, looking into both object level representations and infinitesimal scene elements (3D Gaussians).

Evaluation. Without reliably measuring progress, how can we know if we are actually making progress at all? This is why ensuring accurate, robust and meaningful evaluation procedures is an incredibly important part of building any system or doing any new science. Unfortunately, previous evaluation metrics in the field of Visual Tracking were incredibly limited by only focusing on particular parts of the problem, not the whole, and they could be easily gamed. We seek to address these issues, so that the computer vision community can accurately measure progress in visual tracking.

1.2 Contributions

This thesis contributes to the computer vision literature in four major ways: by proposing a combination of visual tracking with dynamic reconstruction - showing that both can benefit each other to obtain better performance; showing that visual tracking can be combined with differentiable-rendering, novel-view synthesis and volumetric 3D representations in order to obtain incredibly accurate tracking results; proposing a robust and reliable evaluation metric to measure progress fairly and usefully for visual tracking; and finally by extending the notion of visual tracking to an open-world setting where all object need to be tracked, not just those in the set of training classes.

1.3 Background

The prior state of Visual Tracking. At the beginning of the work presented in this thesis in 2018, visual tracking research was in an interesting state. While other computer vision tasks like recognition and detection had seen enormous progress over the few years prior due to the rise and proliferation of deep learning and large training datasets, tracking had not been as thoroughly disrupted and changed, with many classical pre-deep learning methods working similarly to people’s attempts to make deep learning work for tracking.

Traditional approaches were mostly based on simple feature extraction such as histogram-of-oriented-gradients (HOG) features (Dalal & Triggs, 2005) or SIFT features (scale-invariant feature transform) (Lowe, 2004). The whole field of tracking was divided into basically two completely disjoint subsets, with “single-object tracking” (SOT) being the task of tracking a single object for which an example bounding box was given in the first frames, typified by the VOT (Kristan et al., 2021) and OTB (Wu et al., 2013) benchmarks; and “multi-object tracking” (MOT) where all objects from small subset of object classes (e.g. pedestrians and vehicles) needed to be both detected and tracked over time (e.g. the MOTChallenge (Dendorfer et al., 2020b) and KITTI (Geiger et al., 2012) benchmarks).

The first deep learning based methods for each of these sub-tasks were very different. For SOT, most architectures followed a Siamese paradigm, directly comparing (deep) features between the template or previous boxes and a current frame crop, while regressing an output bounding box. Such methods were often trained end-to-end. In contrast MOT methods were mostly based upon the extraction of pre-trained Re-Identification (ReID) embedding vectors, which were then used in a post-hoc way to compare different detections or proposals to associate detections into tracks over time. While these methods were a good start, more progress was needed in order to adequately solve tracking well enough that it could reliably be used in downstream applications.

One of the biggest differences between SOT and MOT was in terms of evaluation. SOT is easy to evaluate, as one only needs to determine if the single predicted box aligns with the single ground-truth box, however MOT is a completely different story, where correct detection of multiple boxes needs to be evaluated while also tacking into account the association of these boxes over time. While previous MOT evaluation solutions (Bernardin et al., 2006; Ristani et al., 2016) were a big step forward compared to what was being used before them, they were still extremely lacking and did not at all adequately measure progress in the field, and as such methods were unable to optimize towards making meaningful progress.

Progression of work in this thesis. My first published work in computer vision was my masters thesis work PReMVOS (Luiten et al., 2018), where I built an approach to tackle video-object segmentation (VOS). After starting my PhD, one of the first

things I worked on was developing the MOTS (Voigtlaender et al., 2019b) (multi-object tracking and segmentation) benchmark with Paul Voigtlaender. This combined the segmentation aspects of VOS that I had previously worked on, with the multi-object tracking setting where multiple objects needed to be both tracked and segmented.

MOTSFusion (Chapter 2) follows directly from this work, in that I wanted to develop a solution for our newly introduced MOTS task that could take advantage of both the pixel-precise segmentations of objects as well as the 3D nature of the world by simultaneously solving both the MOT tasks and the task of dynamic 3D reconstruction of objects.

After working on MOTSFusion, I realized that the way that we (and everyone in computer vision) was evaluating the MOT task was completely broken, and that the evaluation metric results were not indicative of actual method performance. This then led to my HOTA work (Chapter 4), in which I fixed this broken evaluation by developing a novel evaluation procedure accurately measured all relevant aspects of tracking simultaneously.

Open-World Tracking (Chapter 5) follows directly from HOTA, in that after developing HOTA I realised that all tracking evaluation was currently limited to only evaluating a small subset of object classes, and the work in this chapter extended this to evaluating all possible objects in an open world setting.

Finally, the last project of this thesis was Dynamic 3D Gaussians (Chapter 3), which was build both upon the ideas in MOTSFusion (using 3D reconstruction) for tracking, and Open-World Tracking (tracking any object without using any class-specific priors). This extended 3D reconstruction for tracking from only reconstructing dynamic motion at the object level, to reconstructing such motion at the level of infinitesimal scene elements. Also, by focusing on the tracking of these infinitesimal elements in the scene, we are able to move from requiring the notion of “objects” being tracked to densely tracking all elements in a scene. This is then naturally an “open-world setting” where all possible types of scenes and objects are able to be tracked, regardless of what the scene consists of.

1.4 Outline

After this introduction chapter, this thesis is structured as follows:

Chapter 2 - MOTSFusion. In this Chapter we discuss an approach which combines multi-object tracking and dynamic 3D reconstruction. This approach first uses 2D information to build short tracklets, from this 3D reconstructions of each tracklet are created, and these 3D reconstructions and their motion are then used to merge tracklets into longer length tracks and improve tracking results. This is based on the paper “Track to Reconstruct and Reconstruct to Track” (Luiten* et al., 2020).

Chapter 3 - Dynamic 3D Gaussians. In this Chapter we discuss an approach which models the dynamic 3D world a set of 3D Gaussians which can move and rotate through space over time. We develop a way to fit these Gaussians to a multi-camera video capture using differentiable rendering and gradient-based optimization. We show that by adding persistence properties to these Gaussians while also adding in local rigidity regularization results in the motion of these Gaussians following the correct motion of point in 3D space, and thus super accurate tracking emerges simply from trying to solve dynamic reconstruction. This is based on the paper “Dynamic 3D Gaussians: Tracking by Persistent Dynamic View Synthesis” (Luiten et al., 2024).

Chapter 4 - HOTA. In this Chapter we develop an evaluation metric called HOTA for evaluating Multi-Object Tracking (MOT). All previous attempts at evaluating MOT have either almost solely focused on evaluating detection or association separately. Our metric is the first to measure both critical aspects of MOT, as well as the aspect of localization in a principled and balanced way. This enables great progress in multi-object tracking that was not possible before, because now researchers can explicitly see how their approaches improve various different aspects of performance and seek to trade-off different design choices. This is based on the paper “HOTA: A higher order metric for evaluating multi-object tracking” (Luiten et al., 2021).

Chapter 5 - Open-World Tracking. In this Chapter we extend Multi-Object Tracking (MOT) to the open-world. Previously all MOT benchmarks have only focused on an extremely limited number of object classes such as only tracking vehicles and pedestrians. Also because this small number of classes of objects to track was known and fixed beforehand - algorithms could overfit to the specifics of these classes. We ask the question: what would be needed to be able to track any object, even those for which it is not known that they need to be tracked beforehand. To answer this question we propose the Open-World Tracking benchmark, we build a dataset to evaluate this, we design an evaluation metric for evaluating this, and we build up a baseline for tackling this task, ablating and analysing key design decisions in order to determine what are the most important design decisions for bring tracking to the open-world. This is based on the paper “Opening up Open-World Tracking” (Luiten* et al., 2022).

1.5 Additional Works

During this thesis, in addition to the four projects described in Chapters 2-5, I have worked on a large number of further projects. For brevity these are not included in the main content of this thesis, but for completeness these are briefly discussed below:

Object-level Representations for Tracking. In terms of improving the accuracy of tracking approaches, in this thesis I have focused on using 3D representations for

tracking. However the other line of work that I have worked on during this thesis using Object-Level Representations for improving tracking results.

This work started with my Master Thesis work: PReMVOS (Luiten et al., 2018), in which I used off-the-shelf object-level ReID embeddings and developed an algorithm to use these to obtain improved results.

One of the earlier work in this thesis, UnOVOST (Luiten et al., 2020) (together with Idil Esen Zulfikar) extended this idea by using dynamic-programming based algorithms. These ideas were then further extended as the basis of our Open-World Tracking Baselines approach (Luiten* et al., 2022).

Together with Paul Voigtlaender, we built TrackR-CNN (Voigtlaender et al., 2019b) which was one of the first end-to-end approaches for tracking which worked by predicting ReID embeddings and object detections with a single neural network.

We then developed BoLTVOS (Voigtlaender et al., 2019a) and Siam R-CNN (Voigtlaender et al., 2020) together which directly compared pairs of object proposals in a Siamese fashion to determine the likelihood of them belonging to the same object track.

Together with Ali Athar, we then developed the HODOR (Athar et al., 2022) method, which was one of the first to use a transformer architecture for tracking. HODOR took advantage of the flexibility of cross attention to relate and transform embeddings to arbitrarily dimensions in order to simultaneously build both object-level and pixel-level embeddings which could be compared to determine consistent track segmentations across space and time. This architecture also enabled us to train in a weakly supervised manner.

Finally, again with Ali, we developed TarVIS (Athar et al., 2023b), which extended HODOR as a single model that could tackle multiple different version of tracking with the same model. E.g., tracking based on either exemplar guidance, or based on class guidance.

Further Benchmarking of Visual Tracking. In this thesis I focused on the benchmarking efforts of HOTA and Open-World Tracking. However, throughout the duration of this PhD there were several other benchmarks developed.

The first of which was MOTs (Voigtlaender et al., 2019b) in which, together with Paul Voigtlaender, we developed the first Multi-Object Tracking and Segmentation benchmark, extending such a tracking task for the first time to segmentation masks instead of simple bounding boxes.

Together with Valentin Steiner, we developed BEV-KITTI, a benchmark for evaluating segmentation predicted in a bird’s-eye-view representation of the world, directly from standard camera input.

After developing HOTA, I built RobMOTS Luiten (2021) benchmark and challenge, presented at the RVSU workshop that I organised at CVPR 2021. This combined eight different multi-object tracking benchmarks into a singular unified benchmark.

Finally, together with Ali Athar and Achal Dave we developed the BURST benchmark (Athar et al., 2023a), where we annotated Achal’s previous TAO benchmark (Dave

[et al., 2020](#)) with segmentation masks, and repurposed it for evaluating a variety of different tracking tasks.

Improving tasks Adjacent to Tracking. In addition to directly working on Visual Tracking throughout this PhD, I have done a large number of projects that develop functionalities adjacent to tracking.

The first such work was done together with Mark Weber, where we developed Single-Shot Panoptic Segmentation ([Weber et al., 2020](#)), a segmentation approach that worked much faster than all previous methods at the time.

Thomas Jantos and I worked on a detection architecture based on transformers similar to DETR (Detection Transformer) before that paper came out.

Arnie Hoffhues and I worked on an architecture to perform birds-eye-view semantic segmentation.

Neehar Peri and I developed FutureDet ([Peri et al., 2022](#)), an approach to tackle object future trajectory forecasting by re-purposing detectors to directly detect future object locations instead of their current location.

Nikhil Keetha and I developed SplaTAM ([Keetha et al., 2023](#)), an approach to tackle SLAM (simultaneous localization and mapping), e.g. to track the camera pose of a moving camera through a static world by building a 3D map of the world.

MOTSFusion: Track to Reconstruct and Reconstruct to Track

Object tracking and 3D reconstruction are often performed together, with tracking used as input for reconstruction. However, the obtained reconstructions also provide useful information for improving tracking. We propose a novel method that closes this loop, first tracking to reconstruct, and then reconstructing to track. Our approach, MOTSFusion (Multi-Object Tracking, Segmentation and dynamic object Fusion), exploits the 3D motion extracted from dynamic object reconstructions to track objects through long periods of complete occlusion and to recover missing detections. Our approach first builds up short tracklets using 2D optical flow, and then fuses these into dynamic 3D object reconstructions. The precise 3D object motion of these reconstructions is used to merge tracklets through occlusion into long-term tracks, and to locate objects when detections are missing. On KITTI, our reconstruction-based tracking reduces the number of ID switches of the initial tracklets by more than 50%, and outperforms all previous approaches for both bounding box and segmentation tracking.¹

This chapter is based on our paper ([Luiten* et al., 2020](#)), published as a conference paper at ICRA 2020 and as a journal paper in IEEE Robotics and Automation Letters 2020.

¹This paper was conceived, implemented and written together with my student research assistant Tobias Fisher, and we share equal first authorship of this work. Most of the core ideas and code were conceived and implemented during many peer-programming sessions together. Tobias then spent more time running final experiments, while I spent more time writing up, formalizing and describing our ideas. Bastian Leibe contributed discussion and advice along the way.

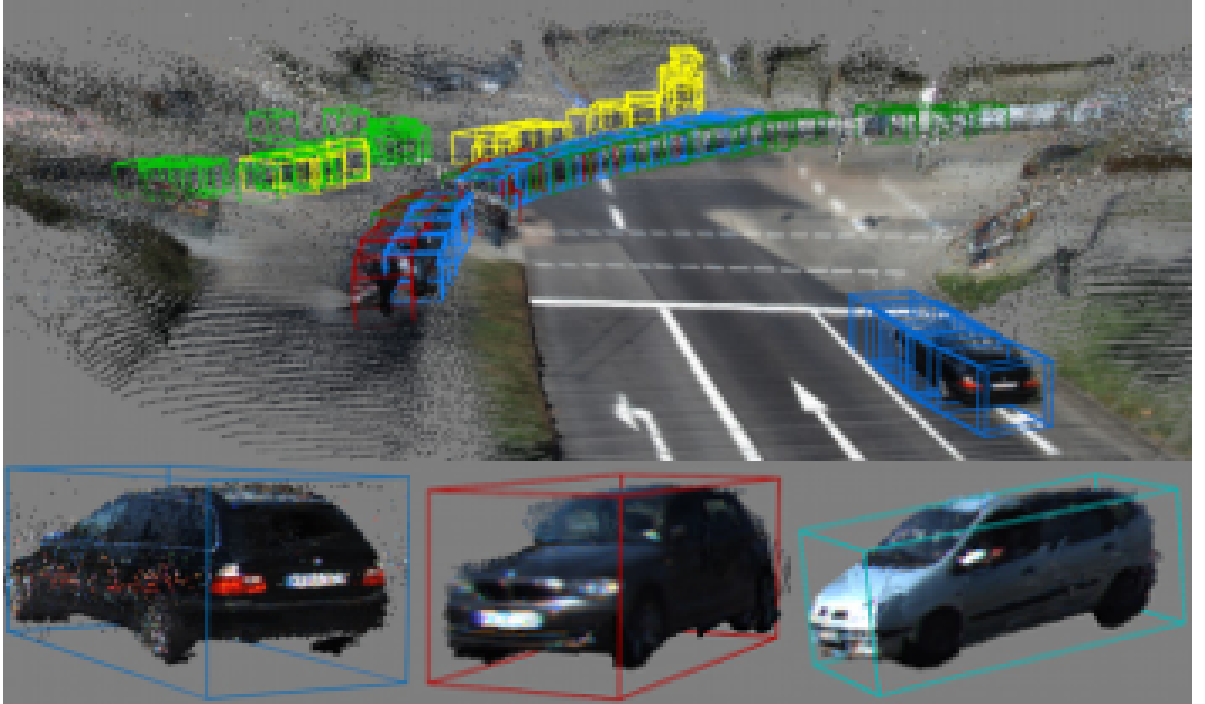


Figure 2.1: Qualitative results of our dynamic 3D reconstruction based tracking. Top: Global scene reconstruction with the 3D bounding boxes for each object track in world coordinates. Bottom: Result of our dynamic object reconstruction, for three of the object tracks in the top image.

2.1 Introduction

Multi-Object Tracking (MOT) is the task of localizing objects in a video and assigning consistent IDs so that each instance of the same object is always given the same ID. This task is crucial for applications such as autonomous vehicles and mobile robots, which need to understand the presence, location and motion of dynamic objects. MOT methods need to track objects both in frames where the objects are contiguously present, as well as through long periods where the objects are not visible due to occlusion. Many current MOT approaches focus on the first part, successfully tracking objects when they are consistently visible, but fail to track objects long-term through disappearance and occlusion, assigning incorrect IDs to objects when they reappear. However, accurate long-term tracking is crucial for understanding complex scenes to the levels required for fully autonomous systems.

To tackle long-term tracking, we propose to use dynamic 3D reconstructions to estimate the 3D motion of objects. Using this motion information, it is possible to track objects through occlusion and to recover missing detections, greatly improving long-term tracking results. Our algorithm, MOTSFusion (Multi-Object Tracking,



Figure 2.2: Qualitative results of our dynamic 3D reconstructions which are used for tracking. For each object, the right visualization shows every 3D point in a given tracklet in world-space. The left visualization shows the 3D reconstruction obtained when mapping each of these points into object-centric space defined by our homogeneous motion transformations. Every point in our set of tracklet masks is present in both visualizations.

Segmentation and dynamic object Fusion), closes the tracking-reconstruction loop, building reconstructions from tracks, and then improving tracking using these reconstructions. An example of our reconstruction-based tracking can be seen in Fig. 2.1.

MOTSFusion consists of a two-stage pipeline. First, we associate detections into short spatio-temporally consistent tracklets in the image domain by calculating a segmentation mask for each detection and measuring the consistency of these masks under a warp defined by optical flow. In a second stage, we project these tracklets into a global 3D domain using the camera ego-motion and a per pixel depth estimate. For each tracklet, a set of homogeneous transformations is calculated which align the object representations at each timestep into a dynamic reconstruction, defining the precise 3D motion of the object. Visual results of this are shown in Fig. 2.2. We extrapolate the 3D trajectories of each tracklet and merge tracklets into long-term object tracks by measuring the consistency of the estimated 3D trajectories. This is able to bridge long periods of occlusion and missing detections. Finally, we fill in missing detections and segmentation masks using the estimated trajectories. Our tracker exploits both 2D image space motion consistency (using optical flow and segmentation masks), and 3D world-space motion consistency (using dynamic 3D reconstructions from depth and ego-motion estimates) for accurate long-term tracking.

In order to fuse dynamic objects in 3D, MOTSFusion relies on the assumption that the objects only move with rigid body transformations. This assumption is valid for cars, but not for pedestrians which can move in an articulated manner. Despite this, our 3D tracklet merging algorithm based on the calculation of these rigid-body transformations performs very well for pedestrians. As visualized in Fig. 2.2, although our method cannot capture the fine-grained details of the articulated motion, it can still accurately capture the overall object motion which is what is required for tracking.

Our long-term tracking is able to reduce the number of ID switches by around 50% compared to the initial 2D tracking results. Furthermore, our method has between 60% and 70% less ID switches than the previous best competing methods (([Sharma et al., 2018](#)) and ([Voigtlaender et al., 2019b](#))), using the same set of detections. This shows that our method successfully creates long-term tracks that are consistent even through occlusion, which is crucial for many applications such as autonomous vehicles. We present several versions of MOTSFusion showing that it can be adapted to work online or offline, and use LiDAR, stereo or monocular depth input.

We make the following contributions. (i) We present a novel long-term tracking pipeline that uses dynamic 3D object reconstructions to merge tracks through long periods of occlusion. (ii) We present a method for recovering missed detections of objects based on their 3D motion obtained from 3D reconstruction. (iii) We present a thorough experimental evaluation validating the effectiveness of our reconstruction-based tracking.

2.2 Related Work

Tracking-by-Detection. Multi-Object Tracking (MOT) has been tackled by many approaches. This work builds upon many successful methods which follow a tracking-by-detection paradigm ([Andriluka et al., 2008](#); [Huang et al., 2008](#); [Milan et al., 2014](#); [Tang et al., 2017](#); [Tian et al., 2018](#)). Such methods can be distinguished by how data-association is performed, and which features are used for association. Data-association for MOT has been approached by network flow ([Zhang et al., 2008](#)), multi-hypothesis tracking ([Kim et al., 2015](#)), quadratic pseudo boolean optimization ([Ess et al., 2008](#)) and conditional random fields ([Choi, 2015](#); [Ošep et al., 2017](#)). Previous methods have exploited the use of 2D motion consistency ([Choi, 2015](#)), 3D motion consistency ([Ošep et al., 2017](#)), and visual embedding similarity ([Luiten et al., 2018](#)) as features for performing association.

Hierarchical 2D/3D Tracking. For data-association, we adopt a hierarchical tracklet creation and merging approach, similar to ([Kaucic et al., 2005](#); [Huang et al., 2008](#)) and ([Geiger et al., 2014](#)), who have shown that such a multi-stage approach can successfully perform tracking. In contrast to these previous approaches, we separate which features are used for association at each stage. First we use only 2D motion consistency for tracklet creation, and then only 3D motion consistency for tracklet merging. Previous approaches ([Geiger et al., 2014](#)) have used appearance models of visual similarity for long-term tracking. We rely solely on 2D and 3D motion consistency, showing that motion cues are enough to perform long-term tracking without exploiting visual similarity.

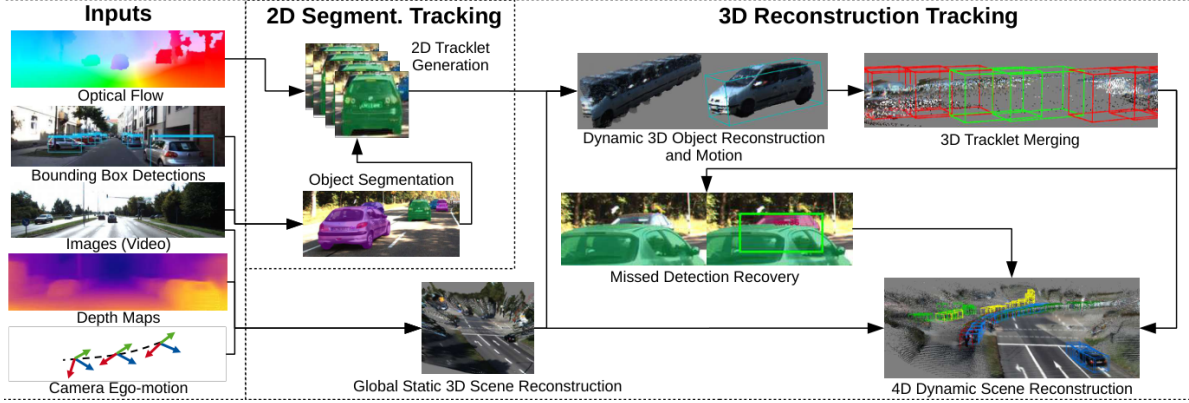


Figure 2.3: Overview of the proposed MOTSFusion method. Given a set of inputs typical for autonomous driving applications, our method performs tracking in a two-stage pipeline. First tracklets are formed using 2D image space motion consistency from optical flow and segmentation masks. Secondly 3D world-space motion consistency is used to merge tracklets together into accurate long-term tracks while recovering missed detections. This is performed by calculating the precise 3D transformations that result in a dynamic 3D object reconstruction for each object.

Segmentation Tracking. Recently segmentation masks have been exploited for improving tracking and creating pixel-accurate tracking results (Luiten et al., 2018; Ošep et al., 2018; Voigtlaender et al., 2019b). These often use optical flow (Ilg et al., 2018) or scene flow (Wedel et al., 2011) to model the motion of each pixel. We adopt both of these techniques using optical flow and scene flow within an object mask to determine motion consistency in both 2D and 3D. We go beyond these methods and use the per-pixel depth values within an the object mask to create 3D object reconstructions over time.

Tracking in 3D. Producing 3D tracking results is invaluable for robotics and autonomous vehicles. There has been a large body of research (Mitzel & Leibe, 2012; Held et al., 2013; Ošep et al., 2017; Luo et al., 2018; Ošep et al., 2018; Sharma et al., 2018) in this area. Most methods (Ošep et al., 2017; Sharma et al., 2018) track using 3D bounding boxes and rely on simple motion cues such as average scene flow vectors. A few methods (Mitzel & Leibe, 2012; Held et al., 2013) have tried to track object reconstructions in 3D by determining the precise transformation for an object between frames. Our method follows a similar idea, however unlike previous methods that used techniques like ICP (Besl & McKay, 1992) for aligning point-clouds, our approach is able to directly optimize for an alignment of points whose correspondences are given by optical flow.

Dynamic Object Reconstruction. Dynamic object reconstruction is closely related to MOT, as to perform reconstruction first tracking needs to be performed. A number of

methods (Bârsan et al., 2018; Runz et al., 2018) have approached this task. Compared to these methods, our method closes the loop; using these reconstructions to further improve tracking.

2.3 Our Approach

MOTSFusion is built upon four key ideas. (i) Accurate short tracklets can be constructed using the 2D motion consistency obtained from segmentation masks and optical flow (Luiten et al., 2018; Voigtlaender et al., 2019b). This results in highly precise short tracklets for when objects are contiguously visible. (ii) For an object undergoing rigid-body transformations, there exists a set of such transformations that fuses all of the point clouds of this object from each timestep into a consistent 3D reconstruction, and these transformations define the 3D motion of that object. (iii) Even for non-rigid objects, if the best-fit rigid-body transformation is calculated in a robust way, this results in 3D motion estimates which are accurate enough to determine the overall 3D motion of an object. (iv) The consistency between the estimated 3D motion of two tracklets contains enough information to determine if these tracklets belong to the same object and should be merged into one longer track.

Following these ideas, we develop MOTSFusion (shown in Fig. 2.3), a two-stage algorithm which first creates short tracklets using the 2D motion consistency of segmentation masks under an optical flow warp, and then fuses these tracklets, using depth and ego-motion estimates, into consistent dynamic 3D object reconstructions. The transformations required for these reconstructions are then used to estimate the 3D motion of object tracklets, merging them into longer tracks if they undergo consistent 3D motion, and using the extrapolated positions to fill in detections where they are missing. MOTSFusion simultaneously tracks objects as both image space segmentation masks and world-space 3D object reconstructions. An example of 3D merging and interpolation is shown in Fig. 2.4. A number of 3D object reconstruction results are shown in Fig. 2.2.

Tracking Inputs. MOTSFusion uses video frames as input, as well as per frame ego-motion and depth estimates. For our main experiments, we use stereo depth (DispNet3 (Ilg et al., 2018)), and ego-motion from a SLAM algorithm (ORB-SLAM2 (Mur-Artal & Tardós, 2017)). However, MOTSFusion is designed to work with any available depth estimates (stereo, LiDAR, RADAR, SfM or monocular) and ego-motion estimates (SLAM, GPS, IMU). We also show results when using LiDAR and single-image depth estimates. For object detection, we use both the recurrent rolling convolution (RRC) detector (Ren et al., 2017) and Track R-CNN (Voigtlaender et al., 2019b). We use optical flow obtained from (Ilg et al., 2018).

Bounding Boxes to Segmentation Masks. We initially estimate a segmentation mask for each bounding box detection. We use a fully convolutional neural network from (Luiten et al., 2018) which we call BB2SegNet. This crops and resizes an image region given by a bounding box to a 385×385 patch and outputs a segmentation mask for each box.

2D Tracklet Generation. We generate tracklets by warping the pixels of each segmentation mask using the optical flow values at these pixels into the next frame. We calculate the IoU (intersection over union) of these warped masks and the set of segmentation masks in the next frame to create association similarities. We use the Hungarian algorithm to assign masks to previously existing tracklets. All masks that are not merged into previous tracklets begin new tracklets. A minimum IoU threshold is required to merge masks.

4D Scene Reconstruction. In order to obtain 3D object motion, we need the 3D location of each object in each timestep in a common world-frame. For our world-frame we use the position of the camera in the first frame and create a 4D (3D + time) point cloud of the scene using depth d_t , camera intrinsics $K \in \mathbb{R}^{3 \times 3}$ and camera position matrix $T_t \in \mathbb{R}^{4 \times 4}$ (the homogeneous transformation matrix of the accumulated ego-camera position over time) with the following equations:

$$(x, y, z)^C = ((u, v, 1) \cdot (K^{-1})^\top) \cdot d_t \quad (2.1)$$

$$(x, y, z, 1)^W = (x, y, z, 1)^C \cdot T_t^\top \quad (2.2)$$

where (u, v) is the pixel location at time t , $(x, y, z)^C$ is the 3D point corresponding to pixel (u, v) in the current camera frame and the $(x, y, z)^W$ is the 3D point in the world frame. This 4D scene reconstruction is used for estimating the motion of objects independent of ego-camera motion, and for visualizing 3D tracking (Fig. 2.1).

Dynamic 3D Object Fusion. We calculate a set of rigid-body transformations that warp the set of 3D points for an object tracklet in each time-step into a consistent 3D object reconstruction. We fit a homogeneous transformation from the world-space points within a pixel-mask from time t to the world-space points within the pixel-mask of the same tracklet at time $t + 1$. This transformation is accumulated over all timesteps so that all points in an object tracklet are warped into a consistent 3D reconstruction.

To minimize the influence of incorrect depth estimates and object masks, we filter the set of 3D points used to fit the transformation by calculating the local outlier factor LOF_k (Breunig et al., 2000) using the local reachability density lrd_k of each point p with its k nearest neighbors N_k . The local reachability density uses the reachability distance rd_k , which is defined by the Minkowski distance d_{min} between two points p

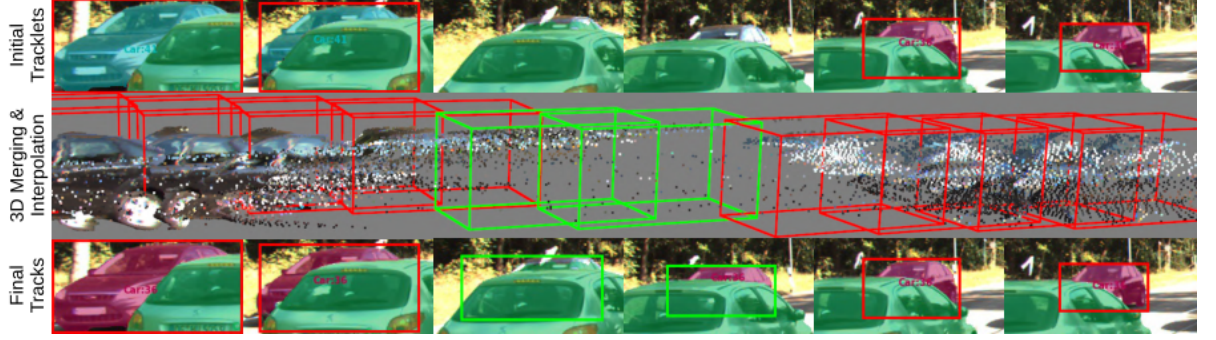


Figure 2.4: Example of our 3D tracklet merging and missing detection filling Top: Initial 2D bounding box detections (red) and segmentation based tracklet results. Middle: Result of our interpolated 3D bounding boxes between the two tracklets, showing the temporal consistency of the two tracklets’ 3D motion through the frames with no detections. Bottom: Original 2D bounding box detections (red) and the new interpolated 2D bounding boxes (green) together with the filled segmentation mask and merged track ID.

and $\mathbf{q}$ and the k -distance d_k of $\mathbf{q}$, which is the maximum of the Minkowski distances of the k nearest neighbors of $\mathbf{q}$, as follows:

$$rd_k(\mathbf{p}, \mathbf{q}) = \max\{d_k(\mathbf{q}), d_{min}(\mathbf{p}, \mathbf{q})\} \quad (2.3)$$

$$lrd_k(\mathbf{p}) := 1 / \left(\frac{\sum_{\mathbf{q} \in N_k} rd_k(\mathbf{p}, \mathbf{q})}{|N_k(\mathbf{p})|} \right) \quad (2.4)$$

Using the local reachability distance of each point, we can compute the local outlier factor LOF_k (Breunig et al., 2000) by:

$$LOF_k(\mathbf{p}) := \frac{\sum_{\mathbf{q} \in N_k} \frac{lrd_k(\mathbf{q})}{lrd_k(\mathbf{p})}}{|N_k(\mathbf{p})|} \quad (2.5)$$

Subsequently, we filter out points with a higher LOF_k than the median LOF_k of all points. For the remaining 3D points at time t and $t + 1$, we use the optical flow vector to correlate points between the two timesteps. We further filter these points to only those for which correspondences exist between the two timesteps. We then sample a maximum of 200 corresponding points over the mask.

On these points we perform a non-linear least-squares optimization by minimizing the $L2$ distance between each pair of corresponding points in the two point clouds to determine the homogeneous transformation that best aligns the two 3D point clouds. We restrict this transformation to be a 3-DoF homogeneous transformation (X and Z translation, and rotation around the ground plane normal). This simplification is

valid for KITTI as the ground-plane is approximately flat. We parametrize the 3D rigid motion with $\xi \in \mathfrak{se}(2)$, the Lie algebra associated with $SE(2)$, which is a minimal representation for this motion. This transformation gives the precise alignment (motion) of an object between timesteps in world coordinates. For each tracklet, we calculate this transformation for all pairs of neighboring frames and accumulate the transformations, merging the point clouds from every timestep into one 3D reconstruction in a consistent object-centric reference frame. Results of this are shown in Fig. 2.2, where it is evident that our transformations are accurate, even when accumulated over many timesteps, and even for objects undergoing non-rigid transformations.

3D Tracklet Merging. Tracklets are merged by analyzing their 3D consistency under the motion given by the object fusion. We examine merging candidates for a terminated tracklet up to N frames beyond the terminated tracklet’s temporal extent. The inherent uncertainty of the motion estimates is measured by the alignment accuracy of the rigid-body transformations using the residual error of the non-linear fitting. We define a “Trusted Motion Region” (TMR) as the set of contiguous motion transformations closest to the end of a tracklet which are all below a residual error threshold (requiring at least two transformations contiguously). We map each of the transformations $\xi = (\xi_x, \xi_z, \xi_\theta)$ in the TMR from global coordinates into object-centric coordinates $\hat{\xi}$ which are centered at the current object center $\mathbf{p}$ as follows:

$$\begin{pmatrix} \hat{\xi}_x \\ \hat{\xi}_z \\ \hat{\xi}_\theta \end{pmatrix} = \begin{pmatrix} \cos(\xi_\theta) - 1 & -\sin(\xi_\theta) & \xi_x \\ \sin(\xi_\theta) & \cos(\xi_\theta) - 1 & \xi_z \\ 0 & 0 & \xi_\theta \end{pmatrix} \begin{pmatrix} \mathbf{p}_x \\ \mathbf{p}_z \\ 1 \end{pmatrix} \quad (2.6)$$

This object-centric motion parametrization $\hat{\xi}$ encodes the same transformation as the original ξ , but now is centered on the moving object center. This allows us to meaningfully average these transformations, as well as to extrapolate them into the future, always relative to the current object center estimate. It is not possible to meaningfully average or extrapolate homogeneous transformations in global coordinates, they must first be transformed into an object-centric coordinate system.

We use the median location of our filtered 3D point set as our object center $\mathbf{p}$. We also model the uncertainty of the object position from stereo estimation. We model an object’s location at each timestep using a multivariate normal distribution. The mean is $\mathbf{p}$ and the covariance, Σ , is obtained using the following (Hartley & Zisserman, 2003):

$$\Sigma = (\mathbf{F}_L^\top(\mathbf{p}) \Sigma_{\text{pix}}^{-1} \mathbf{F}_L(\mathbf{p}) + \mathbf{F}_R^\top(\mathbf{p}) \Sigma_{\text{pix}}^{-1} \mathbf{F}_R(\mathbf{p}))^{-1}, \quad (2.7)$$

where $\mathbf{F}_L(\mathbf{p}), \mathbf{F}_R(\mathbf{p}) \in \mathbb{R}^{2 \times 3}$ are Jacobians (evaluated at $\mathbf{p}$) of the left and right camera projection matrices, respectively. The matrix $\Sigma_{\text{pix}} = (\sigma_u \ 0; 0 \ \sigma_v)$ models uncertainty in the pixel measurements, with σ_u and σ_v both set to 0.5.

The 3D motion consistency between two tracklets is calculated by extrapolating both tracklets' 3D motion towards each other using the average relative transformation of the TMRs into all timesteps between the TMRs of the two tracklets, including the last frame of the earlier TMR and the first frame of the latter. From these 3D position estimates and their uncertainties, the 3D motion consistency of two tracklets is given by the average Mahalanobis distance of these 3D position estimates weighted by their respective uncertainties from the covariances Σ (Eq. 2.7). If the extrapolated motion for one tracklet cannot be estimated robustly due to lack of a TMR, we determine the 3D motion consistency by only extrapolating one towards the other and computing the consistency in the timestep of the last frame of the tracklet for which we have no motion estimate. If both tracklets do not have a robust motion estimate, then both are assumed to be stationary. We merge two tracklets if the 3D motion consistency between them is over a threshold.

Missing Detection Recovery. For each frame between two merged tracklets, we wish to determine if the object is visible and the detection was missed, or if it is actually completely occluded. We first estimate a 3D bounding box for each frame. We use $\mathbf{p}$ as the bounding box center and assume fixed dimensions for pedestrians and cars given by the average width/height/length of the 3D bounding boxes in the KITTI 3D detection training set. This simplifies 3D bounding box estimation, and results in adequate localization for producing segmentations. We set the bounding box orientation θ to the direction of motion if the object undergoes significant motion. Otherwise, we set θ to the direction of the eigenvector with the largest eigenvalue in bird's eye view for all of the 3D points in the object in the current timestep (i.e. the direction of the greatest variance). Examples of 3D bounding boxes can be seen in Fig. 2.1.

We project this 3D bounding box into image space as a 2D bounding box and average it with the previous-frame bounding box corners warped by the median optical flow vector of all points in the bounding box. We run BB2SegNet to get a segmentation mask for this box. We check the validity of this segmentation mask by taking the depth values for the points within the new mask and projecting these back into 3D world coordinates using Eq. 2.1 and Eq. 2.2. If the 3D points of the new mask are sufficiently close to the 3D bounding box center, then the mask passes the consistency check. This check determines if the object has been occluded, since if occluded the points belonging to the new mask will be significantly in front of the estimated 3D bounding box. This is used to fill in missing detections, without introducing many false positives. As well as running missing detection recovery between merged tracklets, we also run it at the beginning and end of whole tracks where we apply this procedure to every frame until we reach a frame where the consistency check fails, the object moves out of the camera field of view, or we reach the end of the video sequence.

Online Version of MOTSFusion. MOTSFusion extrapolates the trajectories of objects both forward and backward in time to perform long-term tracking. Thus, by default

this method works in an offline setting. However, such a reconstruction-based tracking approach can be adapted to work online for use in applications such as autonomous driving and robotics. In order to create an online version of MOTSFusion we simply only extrapolate the trajectories of objects forward in time instead of both forward and backwards. At each new timestep we then match the current detections with the forward extrapolated 3D trajectories, instead of matching the trajectories of two tracklets being extrapolated forward and backward. Finally, for missing detection recovery we also only fill in missing detections in the current frame based on the forward motion estimate. The initial 2D tracklet building step is always performed online.

2.4 Experiments

Datasets. We evaluate MOTSFusion using the KITTI dataset (Geiger et al., 2012), which contains traffic scenes captured from a moving vehicle. We use the KITTI tracking benchmark to evaluate our tracking results for both cars and pedestrians in real-world driving scenes. The annotations for this have been extended with pixel-level mask annotations to evaluate the MOTS task (multi-object tracking and segmentation) (Voigtlaender et al., 2019b). We use the official KITTI test server as well as the validation split from (Voigtlaender et al., 2019b) for evaluation.

Evaluation Metrics. We adopt the CLEARMOT (Bernardin & Stiefelhagen, 2008) metrics which are the standard for KITTI MOT (Geiger et al., 2012). For bounding box tracking methods are ranked by MOTA, which incorporates three error types: false positives (FP), false negatives (FN) and ID switches (IDS). For segmentation tracking we adopt a version of these adapted for masks (Voigtlaender et al., 2019b), where methods are ranked by MOTSA, the segmentation version of MOTA, as well as sMOTSA, which takes into account the segmentation accuracy by incorporating the IoU of the predicted and ground truth masks into the score.

The definition of ID switches is not consistent between benchmarks. For segmentations we use the IDS version from (Voigtlaender et al., 2019b) (also used in MOTChallenge (Leal-Taixé et al., 2015a)). For bounding boxes we present the original version from KITTI (Geiger et al., 2012) as IDS (and MOTA respectively), however it has often been pointed out that this definition does not correctly account for ID switches (Shitrit et al., 2011; Bento & Zhu, 2016; Yu et al., 2016; Maksai et al., 2017). Hence, we also present results on the validation set using the definition from (Voigtlaender et al., 2019b) and (Leal-Taixé et al., 2015a), which we label IDS* and MOTA*, respectively. This definition counts ID switches not only when the ID label switches between two contiguous frames, but also when there is a gap (e.g. occlusion) in between an ID switch occurring.

2 MOTSFusion: Track to Reconstruct and Reconstruct to Track

Tracking	Detect.	Segm.	Speed	sMOTSA	Cars				Pedestrians				
					MOTSA	IDS	FP	FN	sMOTSA	MOTSA	IDS	FP	FN
Ours	RRC	BB2SegNet	0.44	85.7	94.5	31	44	364	-	-	-	-	-
Ours (no fill)	RRC	BB2SegNet	0.43	85.6	94.3	31	37	386	-	-	-	-	-
Ours (2D)	RRC	BB2SegNet	0.14	85.2	94.0	61	37	386	-	-	-	-	-
Ours (online)	RRC	BB2SegNet	0.44	85.5	94.3	35	38	385	-	-	-	-	-
BePix	RRC	BB2SegNet	0.36	84.9	93.8	97	61	337	-	-	-	-	-
Oracle	RRC	BB2SegNet	-	86.9	95.9	0	3	330	-	-	-	-	-
Ours	TrRCNN	BB2SegNet	0.44	82.8	90.5	51	51	661	59.4	72.6	35	99	784
Ours (no fill)	TrRCNN	BB2SegNet	0.43	82.6	90.2	51	41	695	58.8	71.8	35	94	814
Ours (2D)	TrRCNN	BB2SegNet	0.14	81.9	89.6	102	41	695	58.2	71.2	55	94	814
Ours (online)	TrRCNN	BB2SegNet	0.44	82.6	90.2	51	45	688	58.9	71.9	36	95	810
Oracle	TrRCNN	BB2SegNet	-	87.0	96.0	0	19	303	68.7	86.0	0	29	440
Ours	TrRCNN	TrRCNN	0.44	78.2	90.0	36	94	673	50.1	68.0	34	181	855
Ours (no fill)	TrRCNN	TrRCNN	0.43	78.1	89.8	36	86	699	49.5	67.2	34	178	886
Ours (2D)	TrRCNN	TrRCNN	0.14	77.5	89.2	85	86	699	48.9	66.6	53	178	886
Ours (online)	TrRCNN	TrRCNN	0.44	78.1	89.8	44	87	686	49.5	67.3	35	178	882
BePix	RRC	TrRCNN	0.36	76.9	89.7	88	280	458	-	-	-	-	-
TrRCNN	TrRCNN	TrRCNN	0.50	76.2	87.8	93	134	753	46.8	65.1	78	267	822
CIWT	TrRCNN	TrRCNN	0.28	68.1	79.4	106	333	1214	42.9	61.0	42	401	863
CAMOT	TrRCNN	TrRCNN	0.76	67.4	78.6	220	172	1327	39.5	57.6	131	198	1090
Oracle	TrRCNN	TrRCNN	-	82.0	95.1	0	30	361	58.3	80.3	0	23	635

Table 2.1: Mask tracking results on KITTI MOTS Validation. We do not evaluate the RRC detector on pedestrians, because detections are only available for cars. Best result numbers per section in **bold**. Speed is measured in seconds per frame.

Segmentation Tracking: Setup comparison. In Table 2.1 we present results of MOTSFusion for segmentation tracking on the KITTI MOTS validation set. We compare the use of three different combinations of detectors and segmentation methods. We use both the detections and segmentations from TrackR-CNN (Voigtlaender et al., 2019b) for a fair comparison to the previous state-of-the-art. We also use a stronger detector, RRC (Ren et al., 2017), and a stronger segmentation method BB2SegNet (Luiten et al., 2018), in order to compare to the previous state-of-the-art bounding box tracker BeyondPixels (Sharma et al., 2018) with added segmentation masks. We also report results with the detections from TrackR-CNN (Voigtlaender et al., 2019b) with segmentations from BB2SegNet (Luiten et al., 2018). We only evaluate cars when using detections from RRC (Ren et al., 2017) (as pedestrian detections are not available). Using better detections and better segmentations does benefit our method. Using BB2SegNet (Luiten et al., 2018) segmentations instead of TrackR-CNN (Voigtlaender et al., 2019b), sMOTSA increases from 78.2 to 82.8 for cars and from 50.1 to 59.4 for pedestrians (4.6 and 9.3 percentage points). Using RRC (Ren et al., 2017) detections instead of TrackR-CNN (Voigtlaender et al., 2019b), sMOTSA increases another 2.9 percentage points from 82.8 to 85.7 for cars.

Segmentation Tracking: Method ablation. In all five experimental setups, we present ablation results showing how our 3D reconstruction-based tracking improves over the initial tracklets (*Ours* (2D)). In the best setups, with reconstruction-based tracklet

	Cars					Pedestrians				
	sMOTSA	MOTSA	IDS	FP	FN	sMOTSA	MOTSA	IDS	FP	FN
Ours	75.0	84.1	201	295	5342	58.7	72.9	279	465	4870
TrRCNN	67.0	79.6	692	1310	5479	47.3	66.1	481	1179	5363

Table 2.2: Mask tracking results of our best method on KITTI MOTS Test.

merging only (*Ours (no fill)*), we reduce the IDS from 61 to 31 for cars and from 55 to 35 for pedestrians (49% and 36% relative improvement, respectively), while not changing the number of FPs and FNs. We observe similar results over all setups resulting in a sMOTSA increase of between 0.4 and 0.7 percentage points. This is a significant improvement in long-term tracking ability. When also performing missing detection recovery (*Ours*), we reduce the number of FNs from 386 to 364 for cars and from 814 to 784 for pedestrians, while increasing the FPs only marginally from 37 to 44 for cars and from 94 to 99 for pedestrians. Again, we observe similar results over all setups resulting in a further sMOTSA increase of between 0.1 and 0.6 percentage points.

Segmentation Tracking: State-of-the-art comparison. Table 2.1 shows that on the validation set MOTSFusion outperforms TrackR-CNN (Voigtlaender et al., 2019b) with the same detections and segmentations by 2 percentage points for cars (78.2 vs. 76.2) and 3.3 for pedestrians (50.1 vs. 46.8) in sMOTSA, and has 61% (36 vs. 93) and 56% (34 vs. 78) less IDS for cars and pedestrians, respectively. When comparing to BeyondPixels (Sharma et al., 2018) with added segmentations from BB2SegNet, we improve from 84.9 to 85.7 in sMOTSA while reducing the number of IDS from 97 to 31. This is less than a third of the IDS. Using the segmentations from TrackR-CNN (Voigtlaender et al., 2019b) our method outperforms BeyondPixels (Sharma et al., 2018) by 1.3 percentage points in sMOTSA (78.2 vs. 76.9), even though BeyondPixels (Sharma et al., 2018) uses the stronger RRC detector. It also significantly outperforms other tracking methods CIWT (Ošep et al., 2017) and CAMOT (Ošep et al., 2018) using the same detections and segmentations. In Table 2.2 we also evaluate the best versions of MOTSFusion on the MOTS test server, where we significantly outperform the previous best results from (Voigtlaender et al., 2019b).

Segmentation Tracking: Online version. In Table 2.1 we also present results for the online version of MOTSFusion for all five experimental setups. In terms of IDS, the online version works almost as well as the offline version, resulting in negligibly more ID switches over the five setups. This shows the strength of our reconstruction-based tracking approach, even in an online setting. However, the online version cannot leverage the same priors as the offline version for missing detection recovery, where two merged tracklets could have missing measurements in between them, resulting in similar performance to the offline version without detection filling and thus is more reliant on good detections in each frame.

Depth	EPE	Density	sMOTSA	Cars			
				MOTSA	IDS	FP	FN
LiDAR	0.00	4.1%	85.7	94.5	31	44	367
Stereo (Ilg et al., 2018)	1.50	100%	85.7	94.5	31	44	364
Mono (Godard et al., 2019)	2.11	100%	85.3	94.1	51	59	360
None (2D)	-	-	85.2	94.0	61	37	386

Table 2.3: Mask tracking results on KITTI MOTS validation with different depth estimates using RRC (Ren et al., 2017) detections and BB2SegNet (Luiten et al., 2018) segmentations. EPE is the average end-point error compared to the LiDAR depth.

Segmentation Tracking: Oracle comparison. We present results for each setup using an “oracle” tracker, that takes the detections and segmentations, and associates these into tracks using the ground-truth. Our best performing method for cars obtains a sMOTSA only 1.2 percentage points below the theoretical maximum of a perfect tracker. The oracle still has FPs because the MOTS evaluation script constrains all masks in a frame to not overlap, which can cause masks that have been matched to the ground truth to become unmatched.

Segmentation Tracking: Depth input ablation. MOTSFusion works with any depth input, but uses stereo (Ilg et al., 2018) for the main experiments. In Table 2.3 we also provide results using LiDAR measurements and monocular single-image depth estimates (Godard et al., 2019). LiDAR depth is much more accurate than stereo, but it is also very sparse and may suffer from camera-LiDAR calibration issues, whereas monocular depth is usually much less accurate. To illustrate this we compute the end-point error (EPE) of the depth estimates w.r.t. to the LiDAR measurements (for pixels where this is available) and calculate the density of the LiDAR points projected into the image frame. Note that EPE is 0.0 for LiDAR depth because this depth is used as the ground-truth. MOTSFusion performs similarly using LiDAR as when using stereo. This shows that our method is robust to the inaccuracies present in stereo depth, and that it can successfully handle sparse input. When using the much less accurate monocular depth estimates MOTSFusion does not perform as accurately, but still improves over the 2D version which does not use any depth estimation. This shows the robustness of our method to noisy depth input.

Segmentation Tracking: Tracklet merging ablation. Fig. 2.5 shows the results of using different thresholds for the maximum number of frames N between merging candidates. Compared to the initial tracklets (dashed lines), the results quickly and dramatically improve when merging tracklets up to 15 timesteps apart for both the MOTSA score and the number of IDS. When merging beyond 25 frames the missing detection recovery becomes less accurate, introducing more false positives, while the

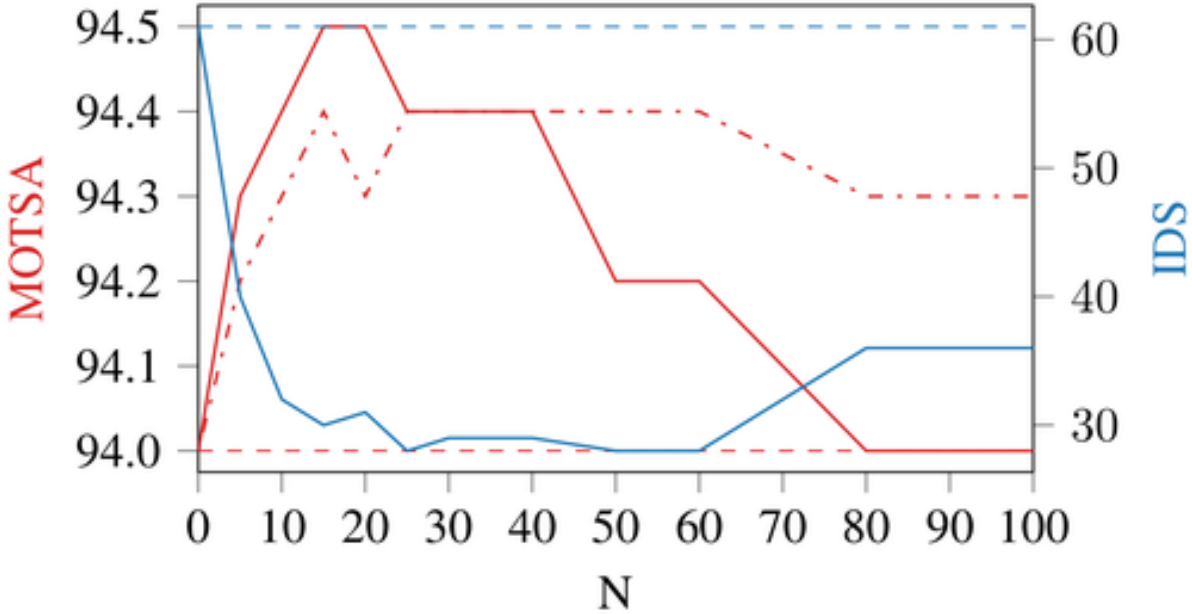


Figure 2.5: Ablation showing the effect of varying the temporal gap size, N , which is the maximum number of frames used for 3D tracklet merging. We present results for both MOTSA (red) and IDS (blue). The dashed lines show the values of our 2D tracklet generation without our reconstruction-based tracking. The dash-dotted line shows the MOTSA when performing tracklet merging but no missing detection recovery.

Tracking	Speed	MOTA	MOTA*	Cars		FP	FN
				IDS	IDS*		
Ours (no fill)	0.43	94.0	93.7	9	31	45	400
Ours (2D)	0.14	93.9	93.3	12	58	45	400
BePix(Sharma et al., 2018)	0.30	93.7	92.9	31	91	88	354
Oracle	-	95.8	95.8	6	6	0	311

Table 2.4: Bounding box tracking results on KITTI Tracking Validation (Cars). The MOTA* and IDS* metrics use the adapted IDS definitions from (Voigtlaender et al., 2019b). All methods use RRC detections (Ren et al., 2017). Speed is measured in seconds per frame.

tracklet merging remains accurate keeping the IDS low. When merging beyond 60 frames the tracklet merging also introduces errors and the IDS increases. For our main experiments we choose a threshold of 20 frames. The gap between the solid red curve and the dash-dotted curve shows how much improvement over the 2D results comes from tracklet merging compared to what comes from missing detection recovery.

Tracking	Detect.	Speed	MOTA	Cars		
				IDS	FP	FN
Ours	RRC (Ren et al., 2017)	0.44	84.83	275	681	4260
BePix (Sharma et al., 2018)	RRC (Ren et al., 2017)	0.30	84.24	468	705	4247
IMMDP (Xiang et al., 2015)	FRCNN (Ren et al., 2015)	0.19	83.04	172	391	5269
PMBM (Scheidegger et al., 2018)	Own	0.01	80.39	121	1007	5616
extraCK (Gunduz & Acarman, 2018)	FRCNN (Ren et al., 2015)	0.03	79.99	343	642	5896
MCCPD (Lee et al., 2016)	FRCNN (Ren et al., 2015)	0.01	78.90	228	316	6713
JCSTD (Tian et al., 2019)	Regio (Wang et al., 2013)	0.07	80.57	61	405	6217
NOMT (Choi, 2015)	Regio (Wang et al., 2013)	0.09	78.15	31	1061	6421
LPSVM (Wang & Fowlkes, 2016)	Regio (Wang et al., 2013)	0.02	77.63	130	1239	6393
SCEA (Yoon et al., 2016)	Regio (Wang et al., 2013)	0.06	75.58	104	1306	6989
CIWT (Ošep et al., 2017)	Regio (Wang et al., 2013)	0.28	75.39	165	954	7345

Table 2.5: Bounding box tracking results on KITTI Tracking Test (Cars). Our method achieves the highest MOTA score of the ten best performing published methods on the KITTI tracking benchmark. Note, that the methods with FRCNN detector train the detector differently. Speed is measured in seconds per frame.

Bounding Box Tracking: State-of-the-art comparison. We evaluate bounding box tracking on a KITTI MOT validation split from (Voigtlaender et al., 2019b) and the official test set. The validation and test results are shown in Table 2.4 and Table 2.5, respectively. MOTSFusion outperforms BeyondPixels (Sharma et al., 2018), the current best performing published method, on both sets by a significant margin, while using the same detections. It also outperforms other state-of-the-art methods that use different detections. For the KITTI MOT definition of IDS, we reduce the IDS compared to (Sharma et al., 2018) from 31 to 9 on the validation split and from 468 to 275 on the test set. When evaluating with the more challenging IDS* on the validation set, we produce only 31 IDS*, compared to 91 for BeyondPixels. MOTSFusion outperforms BeyondPixels on the MOTA metric by 0.3% and 0.6% on the val/test split respectively, while outperforming it by 1.6% on the MOTA* metric on the val set. For bounding box tracking we do not perform missing detection recovery, as the checks that determine if a detection should be filled require checking against a segmentation mask of visible pixels, whereas the bounding boxes in KITTI MOT are “amodal”, which means that they cover both the visible and hidden parts of objects and cannot be validated with our mask-based checks.

Runtime. MOTSFusion runs in 0.44 sec. per frame, including input preprocessing, using a desktop PC with an Intel Core i7-5930K CPU and a GTX 1080 Ti GPU. Input preprocessing takes 0.20 sec. per frame, of which optical flow and depth estimation take 0.07 sec. each, and the relative ego-camera pose takes 0.06 sec. Our 2D tracklet generation takes a total of 0.07 sec. per frame to perform the mask generation, optical-

flow warping and merging into tracklets. The reconstruction-based tracklet merging takes a total of 0.17 sec. of which 0.16 are used to fit the homogeneous transformations for the 3D object fusion, which is the slowest part of our tracker and could easily be sped up in the future using a GPU implementation of least-squares optimization.

In comparison to other trackers on the same hardware, TrackR-CNN (Voigtlaender et al., 2019b) takes 0.50 sec. per frame, and BeyondPixels (Sharma et al., 2018) takes 0.30 sec. per frame (plus 0.06 sec. for mask generation for MOTs). Thus our algorithm is able to run with a similar efficiency while performing much better, especially for long-term tracking. Our method would be faster when using LiDAR or GPS as input because depth and/or ego-motion would come directly from a sensor. As is standard for MOT, the detector runtime is not included for any of the methods.

2.5 Discussion

Contributions. We have presented a framework in which both tracking and 3D object reconstruction can be performed together and can benefit from each other, with tracking enabling reconstruction, and reconstruction enabling long-term tracking through occlusions. In evaluations, our method outperforms previous tracking methods for both bounding box and segmentation tracking. In particular, our approach is able to track objects “long-term” through complete occlusion or missed detections. We demonstrate a clear benefit of using 3D reconstruction to improve tracking, and present a tracking framework, MOTSFusion, where this can be exploited effectively and efficiently.

Connection to other Chapters. This was one of the earliest works done in this thesis. I had just finished working on MOTs (Voigtlaender et al., 2019b), and had already seen from that experience that multi-object tracking could greatly benefit from using explicit 3D representations, particularly for dealing with and tracking through partial and full occlusion. Thus, this paper was born as a way to lift our tracking algorithms to 3D through the view of object level 3D reconstruction. The following two chapters of this thesis are direct follow-ups to the lessons learnt in this work. Chapter 3 builds a higher fidelity 3D representation which is able to capture more complex scenes and motions and can be optimized more robustly. Chapter 4 was born from my experience with this work that the current way to evaluate multi-object tracking were extremely limiting and not able to actually capture progress in the field.

Influence. After being published in ICRA and RA/L in 2020, the paper which makes up the content of this chapter has gone on to have over 140 citations in the scientific literature, and the github repository containing the code for this project has over 180 stars. This work has become one of the de-facto tracking approaches that combine

both image and depth information for multi-object tracking in autonomous driving scenarios, with many follow up works building directly upon the ideas presented here.

Dynamic 3D Gaussians: Tracking by Persistent Dynamic View Synthesis

We present a method that simultaneously addresses the tasks of dynamic scene novel-view synthesis and six degree-of-freedom (6-DOF) tracking of all dense scene elements. We follow an analysis-by-synthesis framework, inspired by recent work that models scenes as a collection of 3D Gaussians which are optimized to reconstruct input images via differentiable rendering. To model dynamic scenes, we allow Gaussians to move and rotate over time while enforcing that they have persistent color, opacity, and size. By regularizing Gaussians' motion and rotation with local-rigidity constraints, we show that our Dynamic 3D Gaussians correctly model the same area of physical space over time, including the rotation of that space. Dense 6-DOF tracking and dynamic reconstruction emerges naturally from persistent dynamic view synthesis, without requiring any correspondence or flow as input. We demonstrate a large number of downstream applications enabled by our representation, including first-person view synthesis, dynamic compositional scene synthesis, and 4D video editing.^{1 2}

This chapter is based on our paper (Luiten et al., 2024), published at the 2024 International Conference on 3D Vision.

3.1 Introduction

Persistent dynamic 3D world modeling would be transformative for both discriminative and generative artificial intelligence. On the discriminative side, this would enable a metric-space reconstruction of every part of the scene over time. Modeling where everything currently is, where it has been, and where it is moving, is crucial for many

¹This paper was conceived, implemented and written solely by myself, with discussion and advice along the way from my three coauthors Georgios Kopanas, Bastian Leibe, and Deva Ramanan.

²See the project website for video results: <https://dynamic3dgaussians.github.io/>

applications including robotics, augmented reality and self-driving. In generative AI, such models could enable new forms of content creation such as easily controllable and editable high resolution dynamic 3D assets for use in movies, video games or the meta-verse. Many such applications require scalable approaches that can be run on high-resolution imagery in real-time. Thus far, no approach has been able to produce photo-realistic reconstructions of arbitrary dynamic scenes with highly-accurate tracks and visually-appealing novel-views, all while being able to be trained quickly and rendered in real-time.

In this chapter we present such an approach by simultaneously tackling the discriminative tasks of dynamic 3D scene reconstruction and dense non-rigid long-term 6-DOF scene-tracking, while addressing the generative task of dynamic novel-view synthesis. We formulate both of these tasks in an analysis-by-synthesis framework, i.e., we build a persistent dynamic 3D representation of the moving scene that is consistent with all the input observations (images from different timesteps and cameras) and from which tracking emerges as a product of correctly modelling the underlying scene with physically plausible spatial consistency priors.

3D Gaussian Splatting (Kerbl et al., 2023) has recently emerged as a promising approach to modelling 3D static scenes. It represents complex scenes as a combination of a large number of coloured 3D Gaussians which are rendered into camera views via splatting-based rasterization. The positions, sizes, rotations, colours and opacities of these Gaussians can then be adjusted via differentiable rendering and gradient-based optimization such that they represent the 3D scene given by a set of input images. In this chapter we extend this approach from modelling only static scenes to dynamic scenes.

Our key insight is that we restrict all attributes of the Gaussians (such as their number, color, opacity, and size) to be the same over time, but let their position and orientation vary. This allows our Gaussians to be thought of as a *particle-based* physical model of the world, where oriented particles undergo rigid-body transformations over time. In order to reconstruct such particles from raw camera imagery, we exploit the property that at any time, they can be efficiently splatted onto to any camera viewpoint, allowing them to be optimized with an image reconstruction loss. Crucially, particles allow us to operationalize *physical priors* over their movement that act as regularizers for the optimization: a local rigidity prior, a local rotational-similarity prior, and a long-term local isometry prior. These priors ensure that local neighborhoods of particles move approximately rigidly between timesteps, and that nearby particles remain closeby over all timesteps.

Previous approaches to neural reconstruction of dynamic scenes can be seen as either Eulerian representations that keep track of scene motion at fixed grid locations (Cao & Johnson, 2023; Fridovich-Keil et al., 2023; Turki et al., 2023) or Lagrangian representations where an observer follows a particular particle through space and time. We fall in the latter category, but in contrast to prior point-based representations (Abou-Chakra et al., 2022; Zhang et al., 2022), we make use of oriented particles that allow for richer

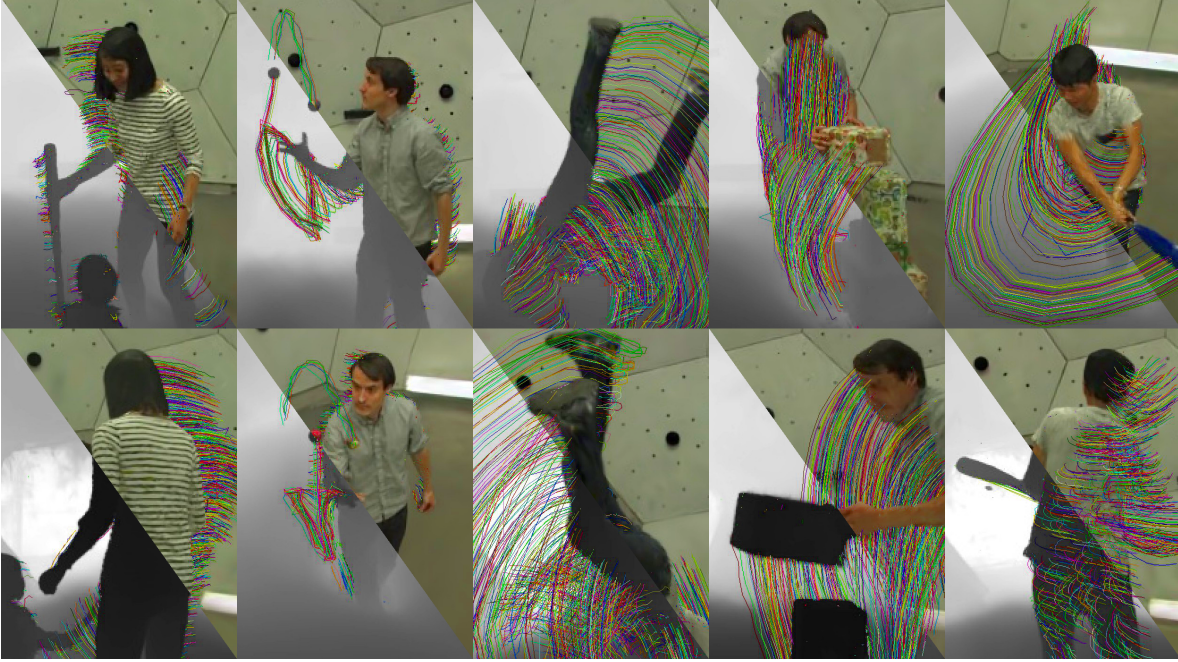


Figure 3.1: Persistent Dynamic Novel-View Synthesis and Tracking Results. Novel-view (unseen) renders of color images and depth maps across 5 scenes (columns) and 2 views (rows) at the same timestep. Each scene is parameterized by 200-300k Dynamic 3D Gaussians which move over time. We render (with occlusions) the 3D trajectories of 2.5% of these over the last 15 timesteps (0.5s).

physical priors (as above) and that directly reconstruct the 6-DOF motion of all 3D points, enabling a variety of downstream applications (see Fig. 3.3 and 3.7).

We perform experiments using synchronized multi-view video (27 training cameras, 4 testing cameras) from the CMU Panoptic Studio dataset (Joo et al., 2015). Our approach achieves 28.7 PSNR on dynamic novel view rendering while rendering at 850 FPS. It is trained on 150 timesteps with 27 training cameras in each timestep in only 2 hours on a single RTX 3090 GPU. Furthermore, our approach results in accurate metric 3D dense non-rigid long-term scene tracking with an average $L2$ error of only 2.21cm in 3D over 150 timesteps, and having an average of 1.57 normalized-pixel error on 2D tracking metrics, which is an order of magnitude (10x) better than previous state-of-the-art. Our method is also able to track the rotation of every 3D point in space, enabling full 6-DOF dense scene tracking. We show visual results in Fig. 3.1.

A remarkable feature of our approach is that tracking arises exclusively from the process of rendering per-frame images. No optical flow, pose skeletons, or any other form of correspondence information is given as input. Due to its persistent and naturally decomposable nature, Dynamic 3D Gaussians are naturally amenable to a number of creative scene editing techniques such as propagating edits over all

timesteps, adding or removing dynamic objects to a scene, or having cameras follow scene elements, as seen in Fig 3.7. Furthermore, the extremely fast rendering and training time make them much easier to work with than previous approaches for dynamic reconstruction (Joo et al., 2014), and enable real-time rendering applications.

3.2 Related Work

In this work, we are interested in solving the combination of dynamic novel-view synthesis, long-term point-tracking and dynamic reconstruction in a unified analysis-by-synthesis framework.

Dynamic Novel-View Synthesis. The general field of novel-view synthesis exploded in popularity with the release of NeRF (Mildenhall et al., 2020) in 2020. Since then there have been a large number of “dynamic NeRF” papers extending the idea of fitting a 3D radiance field to the 4D dynamic domain. Most of these use monocular video as input, whereas we focus on using multi-camera capture. Typically these methods belong to one of the following 5 categories:

(a) Methods that fit a separate representation per timestep, and cannot model correspondence over time (Xian et al., 2021; Bansal & Zollhoefer, 2023).

(b) Methods that represent the scene using an Eulerian representation on a 4D space-time grid, often with various grid decompositions for efficiency such as planar decomposition or hash functions (Cao & Johnson, 2023; Fridovich-Keil et al., 2023; Turki et al., 2023). Such Eulerian approaches also do not give rise to correspondence over time.

(c) Methods that represent the 3D scene in a canonical timestep and use a deformation field to warp this to the rest of the timesteps (Du et al., 2021b; Li et al., 2021; Park et al., 2021b,a; Pumarola et al., 2021; Yang et al., 2022; Liu et al., 2023; Wang et al., 2023). Such methods naturally result in one-directional backward correspondences between each frame and the reference frame, but by default, not the forward correspondences which are required to generate correspondences between any two timesteps. (Gao et al., 2022) advocates for expensive ad hoc root-finding to determine and evaluate such correspondences. A number of methods specifically parameterize their warp-fields in a way that is easily invertible such as linear-blend-skinning (Yang et al., 2022; Song et al., 2023), or reversible neural networks (Wang et al., 2023), in order to obtain correspondences. While this often works quite well (Yang et al., 2022; Song et al., 2023; Wang et al., 2023), requiring a single canonical view for a scene greatly restricts the dynamic representation ability.

(d) Template guided methods (Li et al., 2022; Weng et al., 2022; I. et al., 2023), which model dynamic scenes in restricted environments where the motion can be modelled by a predefined template e.g. a set of human-pose skeleton transformations. This

approach often requires a-priori knowledge of what is to be reconstructed and thus is not a solution for general scenes.

(e) Point-based methods (Abou-Chakra et al., 2022; Zhang et al., 2022), which compared to all of the above categories, hold the most promise for representing dynamic scenes in a way where accurate correspondence over time can emerge due to their natural Lagrangian representation. However, these have not achieved as much attention because the point-based rendering approaches have not worked as well as MLP (Mildenhall et al., 2020) or grid based approaches (Müller et al., 2022).

While a number of view-synthesis works have used Gaussians for static scenes (Keselman & Hebert, 2022; Wang et al., 2022; Kerbl et al., 2023), to the best of our knowledge we are the first to use them to reconstruct dynamic scenes. We build our dynamic Gaussian renderer upon the the static renderer “3D Gaussian splatting” (Kerbl et al., 2023) which is currently the state-of-the-art static scene reconstruction algorithm in terms of both accuracy and speed.

Other than rendering accuracy and speed, modeling the dynamic world with Gaussians has a distinct advantage over points as Gaussian’s have a notion of “rotation” so we can use them to model the full 6 degree-of-freedom (DOF) motion of a scene at every point and can use this to construct physically-plausible local rigidity losses.

Long-Term Point Tracking. Traditionally video tracking algorithms have been focused on tracking whole objects (as bounding boxes or segmentation masks) (Perazzi et al., 2016; Voigtlaender et al., 2019b; Kristan et al., 2021), or tracking dense scene points but only between two timesteps (e.g. optical-flow / scene flow) (Beauchemin & Barron, 1995; Vedula et al., 1999). Recently, a number of approaches have started tackling the task of long-term dense point tracking (Doersch et al., 2022; Harley et al., 2022; Doersch et al., 2023; Karaev et al., 2023; Wang et al., 2023; Zheng et al., 2023), where every pixel in a video needs to be tracked across every video frame. In this chapter, we extend this long-range tracking task to 3D and evaluate it in a multi-camera capture setup. Most of these prior approaches (Doersch et al., 2022; Harley et al., 2022; Doersch et al., 2023; Karaev et al., 2023; Zheng et al., 2023) are deep learning based and work by being trained on a large-dataset of ground-truth point tracks (often using synthetic training data). The most similar method to ours is OmniMotion (Wang et al., 2023) which also fits a dynamic radiance field representation using test-time optimization for the purpose of long-term tracking. They focus on monocular video while we focus on multi-camera capture and as such we can reconstruct tracks in metric-3D while they produce a “pseudo-3D” representation. However, the largest and most significant difference between our method and OmniMotion is that they require dense optical-flow input between every pair of timesteps as an optimization target, which is incredibly expensive as it scales with the number of frames squared. The optical flow estimates already provide a (noisy and inconsistent) tracking result which is made consistent through dynamic modelling of the scene. In contrast, our method

takes no correspondences at all as input and tracking emerges from fitting a persistent representation to the input frames along with physically-based priors.

Dynamic Reconstruction. In addition to the radiance-field based approaches listed above, a number of other approaches have tackled the task of “dynamic reconstruction” (Joo et al., 2014; Newcombe et al., 2015; Slavcheva et al., 2017; Bârsan et al., 2018; Luiten* et al., 2020; Prokudin et al., 2023). Such approaches either rely on the presence of accurate depth cameras (Newcombe et al., 2015; Slavcheva et al., 2017), assume ground-truth object point clouds as input (Prokudin et al., 2023), or are heavily specific to certain domains e.g. reconstructing moving cars in driving scenes (Bârsan et al., 2018; Luiten* et al., 2020). The closest approach to ours is (Joo et al., 2014) which also predicts long-term 3D point tracks and also uses the Panoptic Studio data for evaluation. However this approach requires 480 input cameras (in contrast we use 27) and also requires as input pre-computed optical-flow correspondences, which it then lifts into 3D trajectories.

3.3 Method

Overview. Given a set of images from different timesteps and different cameras ($\mathcal{I}_{t,c}$), along with each camera’s respective intrinsic (K_c) and extrinsic ($E_{t,c}$) matrices, our approach reconstructs the dynamic 3D scene ($\mathcal{S}$) observed by these cameras in a temporally persistent manner.

This reconstruction is performed via test-time optimization and no further training data other than the test scene is used. The reconstruction is performed temporally online, i.e., one timestep of the scene is reconstructed at a time with each one being initialized using the previous timestep’s representation. The first timestep acts as an initialization for our scene where we optimize all properties, and then fix all for the subsequent timesteps except those defining the motion of the scene. Each timestep is optimized via gradient based optimization using a differentiable renderer ($\mathcal{R}$) to render the scene at each timestep into each of the training cameras.

$$\hat{\mathcal{I}}_{t,c} = \mathcal{R}(\mathcal{S}_t, K_c, E_{t,c})$$

The renderings $\hat{\mathcal{I}}_{t,c}$ are compared to the input images $\mathcal{I}_{t,c}$, and the parameters of $\mathcal{S}$ are iteratively updated using automatic differentiation in order to decrease the error between $\hat{\mathcal{I}}_{t,c}$ and $\mathcal{I}_{t,c}$. After convergence, the representation $\mathcal{S}_t$ is a 3D reconstruction of the scene given by each of the training cameras $\{\mathcal{I}_{t,c}, K_c, E_{t,c}\}$ for this timestep. By choosing a suitable representation for $\mathcal{S}$ and applying physically-based regularization losses during optimization, we can ensure that all the $\mathcal{S}_t$ are temporally consistent with one another and that there exists a one-to-one correspondence between every 3D point

in every timestep, along with their corresponding changes in 3D rotation. In this way, tracking emerges from persistent dynamic view synthesis.

Dynamic 3D Gaussians. Our dynamic scene representation ($\mathcal{S}$) is parameterized by a set of Dynamic 3D Gaussians, each of which has the following parameters:

1. a 3D center for each timestep (x_t, y_t, z_t) .
2. a 3D rotation for each timestep parameterized by a quaternion (qw_t, qx_t, qy_t, qz_t) .
3. a 3D size in standard deviations (consistent over all timesteps) (sx, sy, sz)
4. a color (consistent over all timesteps) (r, g, b)
5. an opacity logit (consistent over all timesteps) (o)
6. a background logit (consistent over timesteps) (bg)

This gives a total of $7t + 8$ parameters for each Gaussian. In our experiments, scenes are represented by between 200-300k Gaussians, of which only 30-100k usually are not part of the static background. While the code contains the ability to represent view-dependent color using spherical harmonics, we turn this off in our experiments for simplicity.

Each Gaussian can be thought of as softly representing an area of 3D physical space which is occupied by solid matter. Each Gaussian influences a point in physical 3D space (p) according to the standard (unnormalized) Gaussian equation weighted by its opacity:

$$f_{i,t}(p) = \text{sigm}(o_i) \exp \left(-\frac{1}{2} (p - \mu_{i,t})^T \Sigma_{i,t}^{-1} (p - \mu_{i,t}) \right)$$

Where $\mu_{i,t} = [x_{i,t} \ y_{i,t} \ z_{i,t}]^T$ is the center of each Gaussian i at timestep t , and $\Sigma_{i,t} = R_{i,t} S_i S_i^T R_{i,t}^T$ is the covariance matrix of Gaussian i at timestep t , given by combining the scaling component $S_i = \text{diag}([sx_i \ sy_i \ sz_i])$, and the rotation component $R_{i,t} = \text{q2R}([qw_{i,t} \ qx_{i,t} \ qy_{i,t} \ qz_{i,t}])$, where $\text{q2R}()$ is the formula for constructing a rotation matrix from a quaternion. $\text{sigm}()$ is the standard logistic sigmoid function.

Each Gaussian's influence (f) is both inherently local (being able to represent a small area of space), while also theoretically having infinite extent, such that gradients can flow to them even from a long distance, which is crucial for gradient-based tracking-by-differentiable rendering, as Gaussians which may currently be in the wrong 3D location need to get gradients pushing them towards moving to the correct 3D location through the differentiable renderer. The softness of this Gaussian representation also means that Gaussians typically need to significantly overlap in order to represent a physically solid object. As well as physical density, each Gaussian contributes its own color (r, g, b) to each of the 3D points it influences.

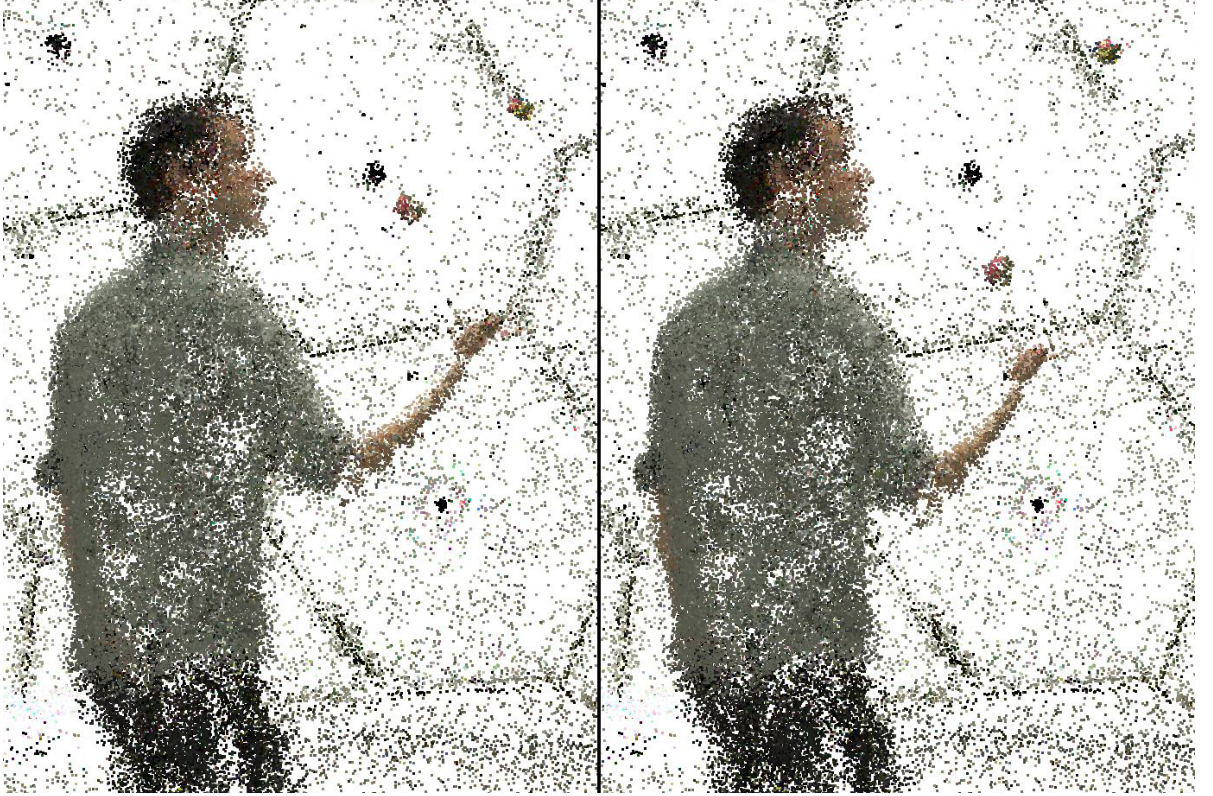


Figure 3.2: Gaussian Centers. Point-cloud of colored centers in contiguous timesteps, showing how they model scene geometry and move over time.

By fixing the size/opacity/color of the Gaussians across time, each Gaussian should represent the same physical aspect of space, even as this space dynamically moves through time. To represent this motion, each Gaussian has a center location and rotation that can move with time, enabling full dense non-rigid 6-DOF tracking of a whole scene.

We visualize the trajectories of these Gaussians in Fig 3.1, the locations of the Gaussian’s centers in Fig 3.2, and the change in rotation over time in Fig 3.3.

Differentiable Rendering via Gaussian 3D Splatting. In order to optimize the parameters of our Gaussians to represent the scene, we need to render the Gaussians into images in a differentiable manner. In this work we use the differentiable 3D Gaussian renderer from (Kerbl et al., 2023) and extend its use to dynamic scenes. This works by splatting 3D Gaussians into the image plane by approximating the projection of the integral of the influence function f along the depth dimension of the 3D Gaussian into a 2D Gaussian influence function in pixel coordinates. The center of the Gaussian is splatted using the standard point rendering formula:

$$\mu^{2D} = K ((E\mu)/(E\mu)_z)$$



Figure 3.3: Relative Rotation Tracking. 1st panel: Left-facing coloured vectors are attached to 3% of Gaussians in the first frame. 2nd and 3rd panels: These vectors move and rotate along with the Gaussians they are attached to, showing that our approach correctly models 6-DOF motion.

where the 3D Gaussian center μ is projected into a 2D image by multiplication with the world-to-camera extrinsic matrix E , z-normalization, and multiplication by the intrinsic projection matrix K . The 3D covariance matrix is splatted into 2D using the formula from (Zwicker et al., 2001):

$$\Sigma^{2D} = JE\Sigma E^T J^T$$

where J is the Jacobian of the point projection formula above, i.e. $\partial\mu^{2D}/\partial\mu$.

The influence function f can now be evaluated in 2D for each pixel for each Gaussian. The influence of all Gaussians on this pixel can be combined by sorting the Gaussians in depth order and performing front-to-back volume rendering using the Max (Max, 1995) volume rendering formula (the same as is used in NeRF (Mildenhall et al., 2020)):

$$C_{\text{pix}} = \sum_{i \in S} c_i f_{i,\text{pix}}^{2D} \prod_{j=1}^{i-1} (1 - f_{j,\text{pix}}^{2D})$$

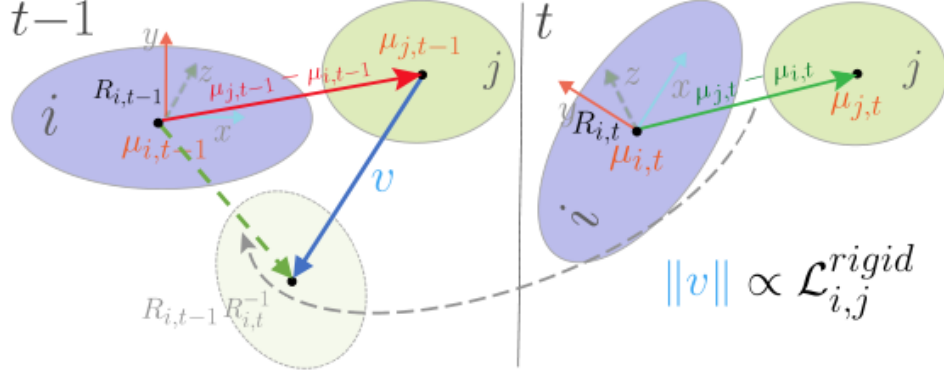


Figure 3.4: Local Rigidity Loss. For each Gaussian i , nearby Gaussians j should move in a way that follows the rigid-body transform of the coordinate system of i between timesteps.

where the final rendered color (C_{pix}) for each pixel is a weighted sum over the colors of each Gaussian ($c_i = [r_i \ g_i \ b_i]^T$), weighted by the Gaussian’s influence on that pixel $f_{i,\text{pix}}^{2\text{D}}$ (the equivalent of the formula for f_i in 3D except with the 3D means and covariance matrices replaced with the 2D splatted versions), and down-weighted by an occlusion (transmittance) term taking into account the effect of all Gaussians in front of the current Gaussian.

The implementation of (Kerbl et al., 2023) uses a number of graphics and CUDA optimization techniques to achieve incredibly fast rendering speeds (e.g., 850 FPS for our scenes), which therefore also enables very fast training.

Physically-Based Priors. We find that just fixing the color, opacity and size of Gaussians is not enough on its own to generate long-term persistent tracks, especially across areas of the scene where there is a large area of near uniform colour. In such situations the Gaussians move freely around the area of similar colour as there is no restriction on them doing so. Since we are trying to model physically moving scenes, it makes sense to look to non-rigid physical modelling for inspiration on how to regularize the optimization procedure to be physically plausible and give correct long-term tracking results. We introduce three regularization losses, short-term local-rigidity $\mathcal{L}^{\text{rigid}}$ and local-rotation similarity $\mathcal{L}^{\text{rot}}$ losses and a long-term local-isometry loss.

The most important of these is the local-rigidity loss $\mathcal{L}^{\text{rigid}}$, defined as:

$$\mathcal{L}_{i,j}^{\text{rigid}} = w_{i,j} \left\| (\mu_{j,t-1} - \mu_{i,t-1}) - R_{i,t-1} R_{i,t}^{-1} (\mu_{j,t} - \mu_{i,t}) \right\|_2$$

$$\mathcal{L}^{\text{rigid}} = \frac{1}{k|\mathcal{S}|} \sum_{i \in \mathcal{S}} \sum_{j \in \text{knn}_{i,k}} \mathcal{L}_{i,j}^{\text{rigid}}$$

This states that, for each Gaussian i , nearby Gaussians j should move in a way that follows the rigid-body transform of the coordinate system of i between timesteps. See Fig 3.4 for a visual explanation.

Since we are performing online optimization, all of $\mu_{i,t-1}$, $R_{i,t-1}$, $\mu_{j,t-1}$ are fixed, and we are optimizing $\mu_{i,t}$, $R_{i,t}$, $\mu_{j,t}$ to ensure that they match the values in $t-1$ up to the rigid body transformation defined by the change in Gaussian i 's own coordinate system. e.g. if i rotates, then j needs to translate (in its own coordinate system) in a way that is equivalent to rotating around the center of i . This loss also applies the other way, forcing the rotation to match the translation, such that we obtain accurate rotation (6-DOF) tracking for every dense point in space, even though we only optimize to match the rendered images, which is not possible with a point-based representation.

We restrict the set of Gaussians j to be the k -nearest-neighbours of i ($k=20$), and weight the loss by the a weighting factor for the Gaussian pair:

$$w_{i,j} = \exp \left(-\lambda_w \|\mu_{j,0} - \mu_{i,0}\|_2^2 \right)$$

which is an (unnormalized) isotropic Gaussian weighting factor. We set λ_w to 2000, which gives a standard deviation of $\sim 2.2\text{cm}$, and calculate this with the distance between the Gaussian centers in the first timestep and fix it over the rest of the timesteps. This results in the rigidity loss only being enforced locally, while still allowing global non-rigid reconstruction.

The rigidity loss is both necessary and adequate on its own to achieve good results. Since the rigidity loss is applied on all points it is applied in both directions between any pair i and j and thus implicitly enforces i and j to have the same rotation, however we found better convergence if we explicitly force neighbouring Gaussians to have the same rotation over time:

$$\mathcal{L}^{\text{rot}} = \frac{1}{k|\mathcal{S}|} \sum_{i \in \mathcal{S}} \sum_{j \in \text{knn}_{i,k}} w_{i,j} \|\hat{q}_{j,t} \hat{q}_{j,t-1}^{-1} - \hat{q}_{i,t} \hat{q}_{i,t-1}^{-1}\|_2$$

where $\hat{q}$ is the normalized quaternion representation of each Gaussian's rotation, which enables smooth optimization. We use the same set of k -nearest-neighbours and weighting function as before.

We apply $\mathcal{L}^{\text{rigid}}$ and $\mathcal{L}^{\text{rot}}$ only between the current timestep and the directly preceding timestep, thus only enforcing these losses over short-time horizons. This sometimes causes elements of the scene to drift apart, therefore we apply a third loss, the isometry loss, over the long-term:

$$\mathcal{L}^{\text{iso}} = \frac{1}{k|\mathcal{S}|} \sum_{i \in \mathcal{S}} \sum_{j \in \text{knn}_{i,k}} w_{i,j} \left| \|\mu_{j,0} - \mu_{i,0}\|_2 - \|\mu_{j,t} - \mu_{i,t}\|_2 \right|$$

This is a weaker constraint than $\mathcal{L}^{\text{rigid}}$ in that instead of enforcing the positions between two Gaussians to be the same it only enforces the distances between them to be the same.

Optimization Details. Each timestep of a scene is optimized one-at-a time. During the first timestep, all parameters of the Gaussians are optimized. After the first timestep the size, color, opacity, and background logit are fixed and only the position and rotation are updated. Thus, our approach can be seen as first performing static reconstruction of the first frame, followed by long-term dense 6-DOF tracking throughout the remaining frames.

Following (Kerbl et al., 2023), in the first timestep we initialize the scene using a coarse point cloud that could be obtained from running colmap, but instead we use available sparse samples from depth cameras. Note that these depth values are only used for initializing a sparse point cloud in the first timestep and are not used at all during optimization.

We use the densification from (Kerbl et al., 2023) in the first timestep in order to increase the density of Gaussians and achieve a high quality reconstruction. For the rest of the frames the number of Gaussians is fixed and the densification is turned off.

We fit the 3D scene in the first frame for 10000 iterations (around 4 minutes) using 27 training cameras, where each iteration renders a single but complete image. For each timestep after that we use 2000 iterations (around 50 seconds), for a total of 2 hours for 150 timesteps. At the beginning of each new timestep we initialize the estimated Gaussian center positions and rotation quaternion parameters by using forward estimate based on a velocity estimated from the current position minus the previous position, and do the same for the quaternion using normalized quaternions (e.g. we also re-normalize the quaternion representation). We find this to be quite important to getting good results. We also reset the Adam first and second order momentum parameters at the start of each timestep.

In our test scenes the subject’s shirt colours (grey) are very similar to the background. We noticed that often the shirt was being mis-tracked as it was confused with the background, while more contrastive elements like pants and hair were being tracked correctly. To increase the contrast between foreground and background parts of the scene we also render a foreground/background mask and apply a background segmentation loss $\mathcal{L}^{Bg}$ against a pseudo-ground-truth background mask, which we can easily obtain by differencing with an image from the dataset where no foreground objects are presents. We also directly apply a loss that background points should not move or rotate, and restrict the above rigidity, rotation and isometry losses to only operate over foreground points to improve efficiency. This also ensures that these losses are never enforced between foreground parts of the scene and the static floor.

Finally, since we are optimizing over 27 different training cameras, each of which has different camera properties such as white balance, exposure, sensor sensitivity, and color calibration, these factors contribute to variations in color representation across

images. Thus we naively model these differences by simply optimizing a scale and offset parameter for each colour channel for each camera separately. We optimize these only over the first timestep and then fix these for the rest of the timesteps.

Tracking with Dynamic 3D Gaussians. After we have trained our Dynamic 3D Gaussian scene representation we can use it to obtain dense correspondence for any point in a 3D scene. To determine the correspondence of any point in 3D space p across timesteps, we can linearize the motion-space by simply taking the point’s location in the coordinate system of the Gaussian that has the most influence $f(p)$ over this point (or the static background coordinate system if $f(p) < 0.5$ for all Gaussians). There is now a well-defined and invertible one-to-one mapping for all points in space across all timesteps giving dense correspondences.

We can use this same idea to track any 2D pixel-location from any input or novel view into any other timestep or view. To do so we first have to determine the 3D point corresponding to an input pixel. We can render out depth-maps for any view by using the Max (Max, 1995) rendering equation for Gaussians (Kerbl et al., 2023) but replacing the colour component of each Gaussian with the depth of that Gaussian’s center, as seen in Figure 3.1. A pixel’s 3D location can be found via unprojection, the highest influence Gaussian determined, tracked and then projected into a new camera frame.

3.4 Experiments

Dataset Preparation. We prepare a dataset which we call PanopticSports. We take six sub-sequences from the sports sequence of the Panoptic Studio dataset (Joo et al., 2015). Each of our six sequences contain interesting motions and objects which we name them after: juggle, box, softball, tennis, football and basketball. We produce visual results for 3 more sequences, handstand, sway and lift but do not include them in evaluation results as they do not have 3D track ground-truth available. For each sequence we obtain 150 frames at 30 FPS, from the set of the HD cameras. There are 31 cameras, which we split into 27 training and 4 testing cameras (cam 0, 10, 15 and 30 are test). Cameras are temporally aligned and have accurate intrinsics and extrinsics provided. The cameras are positioned roughly in a hemisphere around an area of interest in the middle of a capture studio dome. Fig 3.1 shows an example.

We undistort the images from each camera using the provided distortion parameters, and resize each image to be 640x360. We create an initial point cloud for the first timestep to initialize the Gaussians by taking the points from 10 available depth cameras, synchronized to the current timestep, subsample these depth maps by a factor of 2 in both dimensions and obtain an initial colour by projecting these points into the nearest training camera. Points that do not project into any training camera are discarded. We obtain pseudo-ground-truth foreground-background segmentation

Task	Metrics	Method	Juggle	Boxes	Softball	Tennis	Football	Basketball	Mean
View Synthesis	PSNR $\uparrow$	3GS-O	28.19	28.74	28.77	28.03	28.49	27.02	28.21
		Ours	29.48	29.46	28.43	28.11	28.49	28.22	28.7
	SSIM $\uparrow$	3GS-O	0.91	0.91	0.91	0.90	0.90	0.89	0.90
		Ours	0.92	0.91	0.91	0.91	0.91	0.91	0.91
	LPIPS $\downarrow$	3GS-O	0.15	0.15	0.14	0.16	0.16	0.18	0.16
		Ours	0.15	0.17	0.19	0.17	0.19	0.18	0.17
3D Tracking	3D MTE $\downarrow$	3GS-O	32.81	39.95	64.94	75.54	45.57	76.71	55.9
		Ours	1.90	1.97	2.02	2.33	2.45	2.56	2.21
	3D $\delta\uparrow$	3GS-O	13.6	3.5	5.9	4.2	9.8	3.5	6.8
		Ours	77.2	75.9	70.3	69.0	69.4	66.3	71.4
	3D Surv $\uparrow$	3GS-O	56.3	60.8	37.2	16.9	59.6	31.9	43.8
		Ours	100	100	100	100	100	100	100
2D Tracking	2D MTE $\downarrow$	3GS-O	23.86	29.88	51.6	58.15	35.15	64.29	43.8
		PIPS	5.76	8.42	13.3	21.0	23.2	22.6	15.7
		Ours	1.54	1.42	1.69	1.36	1.48	1.93	1.57
	2D $\delta\uparrow$	3GS-O	17.1	10.5	8.9	6.5	15.0	7.2	10.9
		PIPS	55.9	39.5	37.0	28.4	43.5	33.2	39.6
		Ours	80.4	82.5	77.3	80.2	79.7	73.9	78.4
	2D Surv $\uparrow$	3GS-O	71.3	74.4	42.7	23.0	69.6	47.1	54.7
		PIPS	91.6	61.3	88.6	72.2	79.8	77.6	79.0
		Ours	100	100	100	100	100	100	100

Table 3.1: Results on our prepared PanopticSports dataset. See text for details on the dataset, metrics, tasks and methods.

masks by simply doing frame-differencing between each frame and a reference frame from the dataset for each camera when no foreground objects are present.

We prepare ground-truth trajectories for 2D and 3D tracking by taking the high-quality facial and hand key-point annotations that are available for the scene (Simon et al., 2017). For each person in each scene (four scenes have one person, two have two), we take one random face key-point and one random hand key-point from each hand. We manually verify the accuracy of these trajectories by projecting them into the images and viewing them, and remove three that are not accurate. This leaves us with a total of 21 ground truth 3D trajectories.

For 2D tracking, we use the camera-visibility labels to determine if the first point in each 3D trajectory is visible in each camera, and add these videos and projected points to our 2D tracking evaluation. Each 3D point is visible in around 18 cameras for a total of 371 2D ground-truth tracks.

Evaluation Metrics. We evaluate novel-view synthesis on the hold-out 4 camera views across all 150 timesteps for the 6 sequences. We use the standard PSNR, SSIM

and LPIPS metrics (Wang et al., 2004; Zhang et al., 2018). For 2D long-term point tracking we use the metrics from the recent point-odyssey benchmark (Zheng et al., 2023): median trajectory error (MTE), position accuracy (δ), and survival rate. For 3D long-term point tracking there is no prior relevant work, so we decide adapt the 2D metrics from (Zheng et al., 2023) to the 3D domain, except in terms of centimeters in 3-dimensions instead of normalized-pixels in 2D. E.g. MTE is reported as error in cm. δ is calculated at 1, 2, 4, 8 and 16cm thresholds and survival is measuring trajectories that are within 50cm of the ground-truth.

Comparisons. For all three tasks of View-Synthesis, 3D tracking and 2D tracking we compare our Dynamic 3D Gaussian method to the original 3D Gaussian Splatting (Kerbl et al., 2023) which we build upon. We run this in a similar online mode that we run our method for the same number of iteration steps, and thus we call this 3GS-O. We also perform comparisons on the dataset of Particle-NeRF (Abou-Chakra et al., 2022), which includes 20 train and 10 test cameras, and contains much simpler synthetic scenes with both simple geometry and motion. On this benchmark we compare to Particle-NeRF (Abou-Chakra et al., 2022), Instant-NGP (Müller et al., 2022), and TiNeuVox (Fang et al., 2022). The benchmark task for this dataset (Abou-Chakra et al., 2022) is defined as one in which at each timestep methods are allowed a certain number of ray renders and back-propagation steps. Since our approach does not work in the same way we are not able to compare exactly using these criteria. Instead as a reasonably fair criterion we run our method for each timestep for no more wall clock time than Particle-NeRF does, e.g. 200ms per timestep.

For 3D long-term point tracking we have not found any further methods to compare against for this relatively new task. The two methods we would like most to compare to are OmniMotion (Wang et al., 2023) adapted to multi-camera metric space and MAP Visibility Estimation (Joo et al., 2014) since they seem like the most promising competitors. However, since no code is currently available for either we are unable to run such comparisons. For 2D tracking we compare against 2D long-term tracking approaches of which the canonical example is PIPs (Harley et al., 2022). We provide comparison results with PIPs to show a comparison against a learnt method that was trained specifically for long-term 2D point tracking, and is also one of the state-of-the-art approaches for this task. The comparison is not entirely fair to either method. On one hand, our method sees all 27 training cameras while PIPs only sees the one that needs to be tracked in. However, on the other hand, PIPs was trained on 13085 training videos where ground-truth tracks were provided while our method has never seen any ground-truth tracks, nor any other video data other than the camera views for the current test scene. Regardless it is still a good test of our method to compare against such 2D point tracking methods.

PanopticSports Results. We present the results on our prepared PanopticSports dataset in Table 3.1. Our approach achieves good scores on all three novel-view

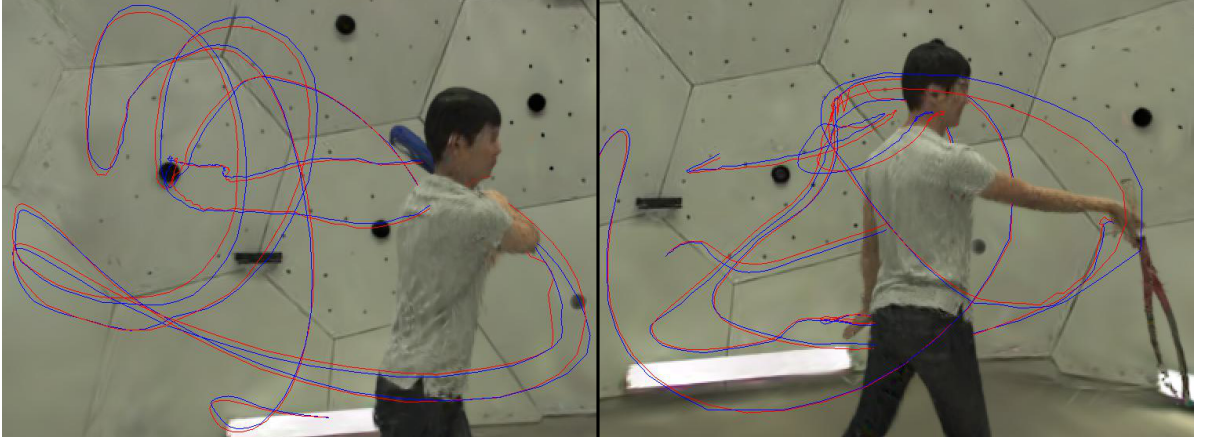


Figure 3.5: Ground-truth Comparison. Comparing our result (blue) to the ground-truth (red). Where ground-truth is noisy, our result may be more accurate.

synthesis metrics, with a final PSNR score of 28.7. Compared to the original 3D Gaussian Splatting (Kerbl et al., 2023) we achieve better PSNR and SSIM scores (although slightly worse LPIPS) across all scenes by correctly modelling the temporal consistency of the dynamic scene. Visual examples of the high-quality novel-view synthesis results can be found in Fig. 3.1. In terms of 3D tracking, our method achieves outstanding results with a median trajectory error of only 2.21cm across all trajectories in all scenes. This is less than the width of a wrist across 150 timesteps of 3D tracking through extremely complex and fast motions (see Fig. 3.1). Our method also has 100% survival rate across all sequences never losing the point to track and an accuracy value of 71.4. The original Gaussian Splatting (Kerbl et al., 2023) approach does not correctly track points in 3D at all with a much higher 55.9cm median trajectory error. A visual comparison of our method compared to the ground-truth can be found in Fig 3.5.

When measuring the 2D tracking ability of our method and comparing it to the 2D tracker PIPs (Harley et al., 2022) we can see where our method really shines. Although it is not a 1-to-1 fair comparison, by comparing our numbers against the numbers of a state-of-the-art tracker we can accurately gauge the performance of our approach. We achieve a 10x lower median trajectory error of only 1.57 pixels compares to PIPs 15.7, have a much higher trajectory accuracy of 78.4 compared to 39.6, and a 100% survival rate compared to 79%.

Overall these results show that our method performs excellently on the tasks of novel-view-synthesis, as well as both 2D and 3D tracking.

Particle-NeRF Dataset Results. On the Particle-NeRF dataset our method achieves almost perfect scores for all of PSNR, SSIM and LPIPS. This is due to the dataset being relatively simple in terms of simple synthetic objects and simple motions. A visual

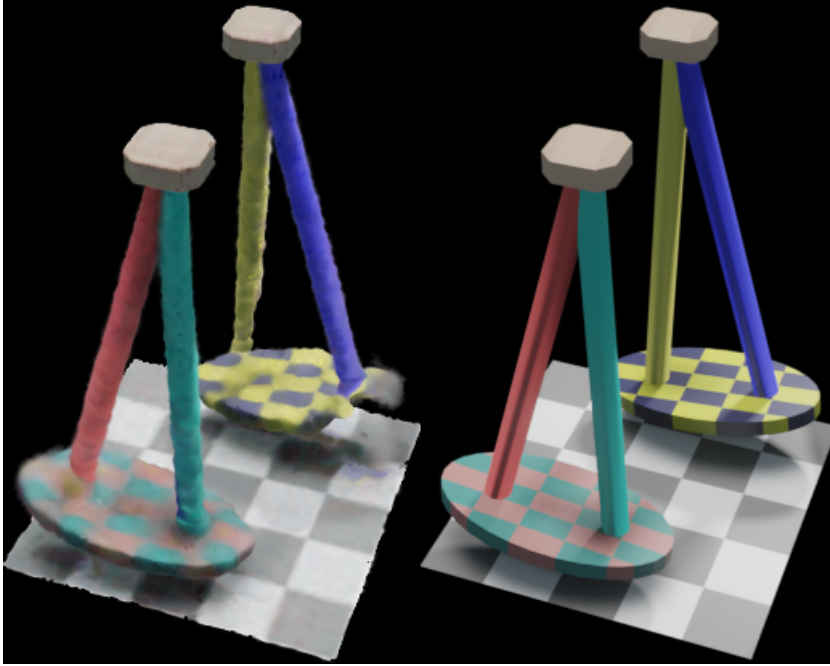


Figure 3.6: Visual comparison. Comparing Particle-NeRF (left) and Ours (right) on the Particle-NeRF dataset.

Method	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$
TiNeuVox-S (Fang et al., 2022)	26.64	0.92	0.14
TiNeuVox (Fang et al., 2022)	27.28	0.91	0.13
InstantNGP (Müller et al., 2022)	24.69	0.91	0.12
Particle-NeRF (Abou-Chakra et al., 2022)	27.47	0.94	0.08
Ours	39.49	0.99	0.02

Table 3.2: Result on the Particle-NeRF dataset. See text for details on the dataset, metrics, tasks and methods.

comparison between our method’s results and those of Particle-NeRF can be seen in Fig 3.6.

Ablation Study. In Table 3.3 we show results on the Juggle sequence of our Panoptic Sports dataset ablating the various different components of our approach. We identify 6 key components of our approach which are above and beyond that of the original 3D Gaussian splatting approach (Kerbl et al., 2023), as described in Section 3.3. We evaluate the effect of removing each of these components from our final method one-at-a-time as well as the effect of removing them all at once, which is then just the method from (Kerbl et al., 2023) run in an online mode over different timesteps.

Exp #	Description	Additions						View Synthesis			3D Tracking		2D Tracking	
		$\mathcal{L}_{\text{Rigid}}$	$\mathcal{L}_{\text{Rot}}$	$\mathcal{L}_{\text{Iso}}$	$\mathcal{L}_{\text{Bg}}$	Fix	Prop	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	3DMTE $\downarrow$	3D $\delta\uparrow$	2DMTE $\downarrow$	2D $\delta\uparrow$
0	Ours - Full	✓	✓	✓	✓	✓	✓	29.48	0.92	0.15	1.90	77.2	1.54	80.4
1	No $\mathcal{L}_{\text{Rigid}}$	✗	✓	✓	✓	✓	✓	28.51	0.91	0.17	4.32	55.2	3.80	58.7
2	No $\mathcal{L}_{\text{Rot}}$	✓	✗	✓	✓	✓	✓	29.43	0.92	0.16	1.91	76.6	1.55	79.8
3	No $\mathcal{L}_{\text{Iso}}$	✓	✓	✗	✓	✓	✓	29.36	0.92	0.16	1.93	76.7	1.72	79.3
4	No $\mathcal{L}_{\text{Bg}}$	✓	✓	✓	✗	✓	✓	24.14	0.82	0.34	8.46	60.0	6.40	63.2
5	No Param Fixing	✓	✓	✓	✓	✗	✓	27.14	0.89	0.22	30.7	57.7	19.15	58.8
6	No Forward Prop	✓	✓	✓	✓	✓	✗	28.48	0.91	0.16	6.32	54.87	5.4	57.7
7	3GS-O	✗	✗	✗	✗	✗	✗	28.19	0.90	0.15	32.81	13.6	23.86	17.1

Table 3.3: Ablation results on the Juggle scene of PanopticSports. See text for details on the dataset, metrics, tasks and methods.



Figure 3.7: Augmented reality applications enabled by Dynamic 3D Gaussians. Left: Dynamic objects can easily be removed from scenes, duplicated and added together with other dynamic objects to new scenes. Center Left: Image edits can be lifted to 3D and then automatically propagated across time. Center Right: Camera views can be attached to dynamic Gaussians that move as the scene moves, e.g. first-person view (above) or juggling-ball’s view (below). Right: Objects can be scanned and added to dynamic scenes in a way that follow the scene. E.g. the hat stays correctly on the person with the correct translation and rotation as he does a handstand.

For view synthesis, the original 3GS-O already works extremely well, but by correctly modelling the motion of components in the scene our full method is able to achieve a boost of 1.3 PSNR. For tracking the original does not accurately track the scene at all, but ours performs very accurately. In terms of key-components required for these results, all of the rigidity loss, the background segmentation loss, the colour/opacity/size parameter fixing, and the forward propagation for timestep initialization are key to obtaining both good tracking results and improvement in view-synthesis results. The rotation loss and isometric loss only provide very small improvement in the measured metrics, however visually we found the reconstruction results to be much more coherent and correct with both of these on over turning either off.

Further Applications. Our Dynamic 3D Gaussian approach also lends itself nicely to being used for editing of dynamic 3D scenes. Because Gaussians are independent, subsets of them can easily be added or removed from scenes to create all sort of interesting effects. E.g., creating realistic renders by combining multiple different dynamic components from different scenes and different backgrounds. We can also very easily propagate edits over time. E.g., logos can be added to surfaces in a single frame, and the colors of the Gaussians can be updated to reflect this change. Such edits will automatically propagate to all other frames of a video. Finally, because we are performing full 6-DOF tracking we can take advantage of this for all sorts of visual effects, for example we could put a camera at “first person view” by attaching it to any particular Gaussian and it will follow where that Gaussian moves and rotates over time. The same can be done for adding objects to the scene that can move and rotate along with the Gaussians they are attached to. We show examples of all of these creative applications in Fig 3.7.

3.5 Discussion

Contributions. In this work, we have introduced a novel method for dynamic 3D scene modeling, view synthesis, and 6-DOF tracking that has relevant applications across various domains, including entertainment, robotics, VR and AR. Utilizing Gaussian elements to model dynamic scenes, our approach uniquely captures movements and rotations, consistent with physical properties. The implications of our method extend beyond the immediate results, offering new avenues for real-time rendering and creative scene editing. Our approach, characterized by efficiency and accuracy, sets a promising direction for future research and practical applications in 3D modeling and tracking, underscoring the potential for further innovation in these fields.

Limitations. While our method achieves excellent results it is not without limitations. For example, by design our method is only able to track parts of scenes that are visible in the initial frame. It would completely fail to reconstruct new objects entering the scene. Our method also requires a multi-camera setup and does not work off-the-shelf on monocular video. We believe that these limitations are the seeds of exciting future research directions to build upon and extend our Dynamic 3D Gaussian representation.

Connection to other Chapters. The work in this chapter can be thought of in many ways as a follow up to that in the previous Chapter 2. In that previous chapter a method was developed in order to do dynamic reconstruction of scenes by dividing dynamic scenes up into a series of different objects, and modeling the motion of each object in the scene. From this tracking results could be improved. This chapter takes this one step further in that scenes are divided up into a much higher fidelity representation, with Gaussian elements representing physical particles making up the scene, and

allowing for a much higher fidelity representation of motion. From this super accurate dense tracking directly emerges. All other chapters in this thesis deal with tracking on an object level, whereas this chapter deals with tracking at a dense 3D particle level, where every infinitesimal point in a scene is tracked densely.

Influence. This paper was released to the public as a pre-print in September 2023, around three months before handing in this thesis, and was presented at 3D conference 6 months later in March 2024. As of September 2024 (1 year since the pre-print, 6 months since conference appearance) this paper had achieved wide Twitter and GitHub fame with 580,000 Twitter views and 1,800 likes, and 1,900 stars on GitHub. This paper has also already been cited over 220 times by academic papers that are a follow-up to, and directly build upon, this work. This paper achieved such wide fame, in such a short amount of time, due to the fact that it changed the nature of the fields of Dynamic Reconstruction, View Synthesis and Tracking, for the first time showing that such excellent results are possible with such an approach.

HOTA: A Higher Order Metric for Evaluating Multi-Object Tracking

Multi-Object Tracking (MOT) has been notoriously difficult to evaluate. Previous metrics overemphasize the importance of either detection or association. To address this, we present a novel MOT evaluation metric, HOTA (Higher Order Tracking Accuracy), which explicitly balances the effect of performing accurate detection, association and localization into a single unified metric for comparing trackers. HOTA decomposes into a family of sub-metrics which are able to evaluate each of five basic error types separately, which enables clear analysis of tracking performance. We evaluate the effectiveness of HOTA on the MOTChallenge benchmark, and show that it is able to capture important aspects of MOT performance not previously taken into account by established metrics. Furthermore, we show HOTA scores better align with human visual evaluation of tracking performance. ¹

This chapter is based on our paper ([Luiten et al., 2021](#)), published in the International Journal of Computer Vision, 2020.

4.1 Introduction

Multi-Object Tracking (MOT) is the task of detecting the presence of multiple objects in video, and associating these detections over time according to object identities. The MOT task is one of the key pillars of computer vision research, and is essential for many scene understanding tasks such as surveillance, robotics or self-driving vehicles. Unfortunately, the evaluation of MOT algorithms has proven to be very difficult. MOT is a complex task, requiring accurate detection, localisation, and association over time.

¹This paper was conceived, implemented and written solely by myself, with discussion and advice along the way from my six coauthors Aljosa Osep, Patrick Dendorfer, Philip Torr, Andreas Geiger, Laura Leal-Taixe and Bastian Leibe.

This chapter defines a metric, called HOTA (Higher Order Tracking Accuracy), which is able to evaluate all of these aspects of tracking. We provide extended analysis as to why HOTA is often preferable to current alternatives for evaluating MOT algorithms. As can be seen in Fig. 4.1, currently used metrics MOTA (Bernardin & Stiefelhagen, 2008) and IDF1 (Ristani et al., 2016) overemphasize detection and association respectively. HOTA explicitly measures both types of errors and combines these in a balanced way. HOTA also incorporates measuring the localisation accuracy of tracking results which is not present in either MOTA or IDF1.

HOTA can be used as a single unified metric for ranking trackers, while also decomposing into a family of sub-metrics which are able to evaluate different aspects of tracking separately. This enables clear understanding of the different types of errors that trackers are making and enables trackers to be tuned for different requirements.

The HOTA metric is also intuitive to understand. This can be seen clearly in Fig. 4.1. The detection accuracy, DetA, is simply the percentage of aligning detections. The association accuracy, AssA, is simply the average alignment between matched trajectories, averaged over all detections. The final HOTA score is the geometric mean of these two scores averaged over different localisation thresholds.

In this chapter we make four major novel contributions: (i) We propose HOTA as a novel metric for evaluating multi-object tracking (Sec. 4.5); (ii) We provide thorough theoretical analysis of HOTA as well as previously used metrics MOTA, IDF1 and Track-mAP, highlighting the benefits and shortcomings of each metric (Sec. 4.7 and 4.9); (iii) We evaluate HOTA on the MOTChallenge benchmark and analyse its properties compared to other metrics for evaluating current state-of-the-art trackers (Sec. 4.10); (iv) We perform a thorough user-study comparing how different metrics align with human judgment of tracking accuracy and show that HOTA aligns closer with the desired evaluation properties of users compared to previous metrics (Sec. 4.11).

4.2 Related Work

Early History of MOT Metrics. Multi-Object tracking has a long history dating back to at least the 70s (Singer et al., 1974; Smith & Buechler, 1975; Stein & Blackman, 1975; Reid, 1979). Early work tended to evaluate using their own simple evaluation metrics, such that comparison was not possible between groups. In the early 2000s a number of different groups sought to define standard MOT evaluation metrics. This included the PETS (Performance Evaluation of Tracking and Surveillance) workshop series (Young & Ferryman, 2005), the VACE (Video Analysis and Content Extraction) program (Kasturi et al., 2006), and a number of other groups (Carletta et al., 2005; Nghiem et al., 2007; Waibel et al., 2009). In 2006, the CLEAR (Classification of Events, Activities and Relationships) workshop (Stiefelhagen et al., 2006) brought together all of the above groups and sought to define a common and unified framework for evaluating MOT algorithms. This became the CLEAR MOT metrics (Bernardin & Stiefelhagen, 2008)

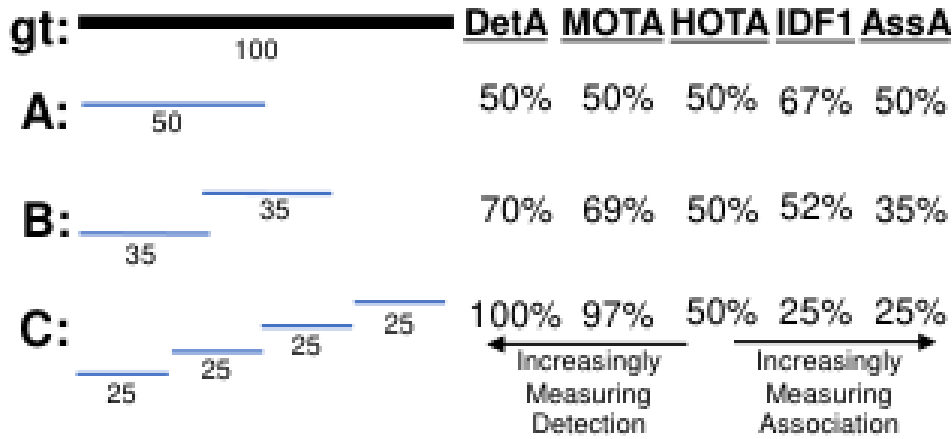


Figure 4.1: A simple tracking example highlighting one of the main differences between evaluation metrics. Three different trackers are shown in order of increasing detection accuracy and decreasing association accuracy. MOTA and IDF1 overemphasize the effect of accurate detection and association respectively. HOTA balances both of these by being an explicit combination of a detection score DetA and an association score AssA.

which positions the MOTA metric as the main metric for tracking evaluation alongside other metrics such as MOTP. MOTA was adopted for evaluation in the PETS workshop series (Ellis & Ferryman, 2010) and remains, to this day, the most commonly used metric for evaluating MOT algorithms, although it has often been highly criticised (Shitrit et al., 2011; Leichter & Krupka, 2013; Milan et al., 2013; Luo et al., 2014; Bento & Zhu, 2016; Ristani et al., 2016; Yu et al., 2016; Leal-Taixé et al., 2017; Maksai et al., 2017; Maksai & Fua, 2019; Wang et al., 2019a; Dave et al., 2020; Dendorfer et al., 2020a; Luiten* et al., 2020) for its bias toward overemphasizing detection over association (see Fig. 4.1), as well as a number of other issues (see Sec. 4.9).

Benchmarks’ use of Metrics. In the last five years the two most commonly used benchmarks for evaluating MOT have been the MOTChallenge (Leal-Taixé et al., 2015b; Milan et al., 2016a; Dendorfer et al., 2019) and KITTI (Geiger et al., 2012) benchmarks. Both of these have ranked trackers using the MOTA metric, contributing to the general widespread use of MOTA in the community.

Within the last few years the multi-object tracking community has grown enormously due in part to large investment from the autonomous vehicle industry. This has resulted in a large number of new MOT benchmarks being proposed. Many of these rank trackers using the MOTA metric (PANDA (Wang et al., 2020), BDD100k (Yu et al., 2020), Waymo (Sun et al., 2020), ArgoVerse (Chang et al., 2019), PoseTrack (Andriluka et al., 2018), MOTS (Voigtlaender et al., 2019b)), or a variation of MOTA (nuScenes (Caesar et al., 2020), UA-DETRAC (Wen et al., 2020)).

Two other metrics have recently been adopted by some MOT benchmarks. The IDF1 metric (Ristani et al., 2016) which was proposed specifically for tracking objects through multiple cameras has been used by “multi-camera MOT” benchmarks such as Duke-MTMC (Ristani et al., 2016), AI City Challenge (Naphade et al., 2017) and LIMA (Layne et al., 2017). IDF1 has also recently been implemented as a secondary metric on the MOTChallenge benchmark, and has become preferred over MOTA for evaluation by a number of single camera tracking methods (Maksai et al., 2017; Maksai & Fua, 2019; Wang et al., 2019a) due to its focus on measuring association accuracy over detection accuracy (see Fig.4.1). IDF1 however exhibits unintuitive and non-monotonic behaviour in regards to detection (see Sec. 4.9).

The Track-mAP metric (also called 3D-IoU) was introduced for tracking evaluation on the ImageNet-Video benchmark (Russakovsky et al., 2015). Recently it has been adapted by a number of benchmarks such as YouTube-VIS (Yang et al., 2019), TAO (Dave et al., 2020) and VisDrone (Zhu et al., 2020). Track-mAP differs from previously described metrics in that it does not operate on a set of fixed tracks, but rather requires a set of tracks ranked by the tracker’s confidence that each track exists. This makes Track-mAP incompatible with many of the commonly used benchmarks (such as MOTChallenge and KITTI) which do not require trackers to output a confidence score with their predictions. Track-mAP suffers from many of the same drawbacks as IDF1 due to its use of global track-based matching, while also having a number of other drawbacks related to the use of ranking-based matching (See Sec. 4.9).

A number of extensions to the MOTA metric, such as PR-MOTA (Wen et al., 2020) and AMOTA (Weng et al., 2020), have previously been proposed to adapt the MOTA metric to handle confidence ranked tracking results as is done in Track-mAP. We present a simple extension to our HOTA metric in Sec. 4.8 which similarly extends HOTA to confidence ranked results, and which reduces to the standard HOTA metric when taking a fixed set of detections above a certain confidence threshold.

A number of other metrics (Smith et al., 2005; Edward et al., 2009; Shitrit et al., 2011; Bento & Zhu, 2016; Rahmathullah et al., 2016; Wu et al., 2017) have been proposed for MOT evaluation, but to the best of our knowledge none of them have been adopted by any MOT benchmarks and thus have not become widely used for evaluation.

Other metrics such as the trajectory-based metrics (Mostly-Tracked, Partially-Tracked, Mostly-Lost, Fragmentation) (Wu & Nevatia, 2006; Li et al., 2009), and the metrics of Leichter and Krupka (False Negative Rate, False Positive Rate, Fragmentation Index, Merger Index, Mean Deviation) (Leichter & Krupka, 2013) are commonly shown as secondary metrics on benchmarks but are never used to comprehensively rank trackers as they are too simple, often focusing on only a single type of error each, and easy to be gamed if desired.

Meta-Evaluation of Metrics. PETS vs VACE (Manohar et al., 2006) discusses the trade-off between presenting multiple evaluation metrics vs a single unifying metric. Their conclusion is that multiple metrics are useful for researchers to debug algorithms

and identify failure components, while a unified metric is useful for end-users wishing to easily choose highly performant trackers from a large number of options. We resolve this conflict by presenting both a unified metric, HOTA, and its decomposition into simple components which can be used to analyse different aspects of tracking behaviour (See Sec. 4.6).

Milan et al. (Milan et al., 2013) analyse the tracking metrics available in 2013 (CLEAR MOT (Bernardin & Stiefelhagen, 2008) and Trajectory-based Metrics (Wu & Nevatia, 2006)) and identify a number of deficits present in these metrics, particularly in MOTA. Based on this analysis they find none of the metrics that were analysed to be suitable as a single unified metric and recommend to present results over all available metrics. We present HOTA as a solution to such issues, as a metric suitable for unified comparison between trackers.

Leichter and Krupka (Leichter & Krupka, 2013) present a theoretical framework for analysing MOT evaluation metrics, which involves two components. The first is a characterisation of five error types that can occur in MOT (False negatives, False positives, Fragmentation, Mergers and Deviation). The second component is the description of two fundamental properties that MOT evaluation metrics should have: monotonicity and error type differentiability. In (Leichter & Krupka, 2013), they show that all previous metrics (including MOTA) do not have either of these properties. They propose a set of five separate simple metrics, one for each error type, however they make no effort to combine these into one unified metric, and thus the usefulness of these metrics in comparing trackers is limited. In Sec. 4.6 we show how HOTA can be decomposed into components which correspond to each of these five error types (Detection Recall, Detection Precision, Association Recall, Association Precision and Localisation Accuracy, respectively), and as such HOTA has the property of error type differentiability which requires that the metrics are informative about the tracker's performance with respect to each of the different basic error types. In Sec. 4.7 we show how the combined HOTA metric is strictly monotonic with regards to each of these five types of errors, thus having the second required property. In Sec. 4.9 we show that no other recently used metric has these desirable properties.

Tracking the Trackers (Leal-Taixé et al., 2017) sought to analyse different evaluation metrics using human evaluators, in order to find out how well different metrics reflect human perception of the quality of tracking results. They only evaluated the set of CLEAR MOT (Bernardin & Stiefelhagen, 2008) and trajectory-based (Wu & Nevatia, 2006; Li et al., 2009) metrics. From these metrics they found that MOTA remains the most representative measure that coincides to the highest degree with human visual assessment, despite pointing out its many limitations. In Sec. 4.11 we seek to repeat this study and compare our HOTA metric with MOTA and IDF1 using human visual evaluation. We find that HOTA performs much better in this user-study than both MOTA and IDF1, particularly among MOT researchers.

4.3 Preliminaries

In this section, we lay the framework needed for understanding the content of this chapter. This includes describing the multi-object task, the role of evaluation metrics, the ground-truth and prediction representation and the notation we will be using, as well as other fundamental concepts that will be used throughout the chapter.

What is Multi-Object Tracking? Multi-Object Tracking (MOT) is one of the core tasks for understanding objects in video. The input to the MOT task is a single continuous video (although we present extensions for our metric for multi-camera tracking in Sec. 4.8), split up into discrete frames at a certain frame rate. Each discrete frame could be an image, or could be another representation such as point cloud from a laser scanner.

The output of the MOT task is a representation that encodes the information about: (a) what objects are present in each frame (detection), where they are in each frame (localisation) and whether objects in different frames belong to the same or different objects (association).

Evaluation Metrics and Ground-Truth. In order to evaluate how well a tracker performs, we need to compare its output to a ground-truth set of tracking results. The purpose of an evaluation metric is to evaluate the similarity between the given ground-truth and the tracking results. This is not a well defined problem, as there are many different ways of scoring such a similarity (especially between complex representations such as sets of trajectories). However, the choice of evaluation metric is extremely important, as the properties of the metric determine how different errors contribute to a final score, as such it is favourable that metrics have certain properties. The choice of metric also has the ability to heavily influence the direction of research within the research community. In the age of competitive benchmarks, a lot of research (for better or for worse) is evaluated on its ability to improve the scores on these benchmarks. If benchmarks are using metrics to evaluate these scores which are biased towards only certain aspects of a task, this will also bias research and methods towards focusing more on these aspects.

MOT Ground-Truth and Prediction Format. The set of ground-truth tracks is represented as a set of detections (gtDets) in each video frame, where each gtDet is assigned an id (gtID), such that the gtIDs are unique within each frame and the gtIDs are consistent over time for detections from the same ground-truth object trajectory (gtTraj).

For most evaluation metrics (MOTA, IDF1 and HOTA) a tracker's prediction is in the same format as the ground-truth data. It consists of a set of predicted detections (prDets) in each frame, each assigned a predicted id (prID), such that the prIDs are unique within each frame and consistent over time for detections from the same predicted object trajectory (prTraj).

For the Track-mAP metric, in addition to the prDets with PrIDs as above, each prTraj is assigned a confidence score estimating how likely it is that this trajectory exists. If each prDet is assigned a confidence score instead of each prTraj, the confidence score for the prTraj is simply the average of the confidence scores over the prDets that belong to the prTraj.

For tracking multiple object classes, each gtTraj and prTraj may also be assigned a class id (gtCl / prCl). Previous metrics have all been applied per class and averaged over classes. Thus for simplicity we can ignore this class id when defining metrics and assume that metrics are calculated only over a single class of objects at a time. However, we also present an extension to HOTA in Sec. 4.8 which explicitly deals with multi-class tracking.

Types of Tracking Errors. The potential errors between a set of predicted and ground-truth tracks can be classified into three categories: detection errors, localisation errors and association errors (Leichter & Krupka, 2013). Detection errors occur when a tracker predicts detections that do not exist in the ground-truth, or fails to predict detections that are in the ground-truth. Association errors occur when trackers assign the same prID to two detections which have different gtIDs, or when they assign different prIDs to two detections which should have the same gtID. Localisation errors occur when prDets are not perfectly spatially aligned with gtDets.

There are other ways of defining basic error types for MOT, such as identification errors (Ristani et al., 2016). However detection, association and localisation errors are the most commonly used error types (Leichter & Krupka, 2013) and are widely applicable for evaluating tracking for a wide range of different tracking scenarios (see Sec. 4.6 for examples). As such we only consider these error types when analysing and comparing evaluation metrics.

Different Detection Representations. Object detections within each frame (gtDets and prDets) may have a number of different representations depending on the domain and application. Commonly used representations include 2D bounding boxes (Geiger et al., 2012; Leal-Taixé et al., 2015b), 3D bounding boxes (Chang et al., 2019; Sun et al., 2020), segmentation masks (Voigtlaender et al., 2019b; Yang et al., 2019), point estimates in 2D or 3D (Ellis & Ferryman, 2010), and human pose skeletons (Andriluka et al., 2018). In general tracking evaluation metrics are agnostic to the specific representation except for the need to define measures of similarity for each representation.

Measures of Similarity. Most metrics require the definition of a similarity score between two detections, $\mathcal{S}$. Track-mAP requires such a similarity score between trajectories, $\mathcal{S}_{tr}$.

This similarity score should be chosen based on the detection representation used, but should be constrained to be between 0 and 1, such that when $\mathcal{S}$ is 1 the prDet and gtDet perfectly align, and when $\mathcal{S}$ is 0 there is no overlap between detections. The most

commonly used similarity metric for 2D boxes, 3D boxes and segmentation masks is IoU_{Loc} , which is the spatial intersection of two regions divided by the union of the regions. For point representations (and human joint locations), a score of one minus the Euclidean distance is often used, such that points are said to have zero overlap when they are more than one meter apart.

Bijjective Matching. A common procedure in MOT evaluation metrics (MOTA, IDF1, HOTA) is to perform a bijective (one-to-one) matching between gtDets and prDets . This ensures that all gtDets are matched to at most one prDet and vice versa, and that any extra or missed predictions are penalised. Such a bijective mapping can be easily calculated by calculating a matching score between all pairs of gtDet and prDet and using the Hungarian algorithm for finding the matching that optimises the sum of the matching score. Usually there is a minimum similarity $\mathcal{S} \geq \alpha$ requirement for a match to occur. After matching has occurred, we have some gtDets and prDets that are matched together. We call these pairs as true positives (TP). These are considered correct predictions. Any gtDets that are not matched (missed) are false negatives (FN). Any prDets that are not matched (extra predictions) are false positives (FP). FNs and FPs are two types of incorrect predictions.

Matching vs Association. The words *match* and *association* are used throughout this chapter to refer to two different things. We refer to a *match* as a pair consisting of a matching ground-truth detection and a predicted detection. On the other hand *association* refers to a number of detections with the same ID such that they are associated to the same trajectory.

Jaccard Index and IoU.. The Jaccard Index is a measure of the similarity between two sets. It is defined as follows:

$$\text{Jaccard Index} = \frac{|\text{TP}|}{|\text{TP}| + |\text{FN}| + |\text{FP}|} \quad (4.1)$$

The Jaccard index is commonly called the IoU (intersection over union) because it is calculated as the intersection of the two discrete sets divided by their union. However, in tracking IoU is also often used for describing the spatial overlap between two spatial regions (e.g., boxes, masks). In order to not confuse these terms we use “Jaccard index” to refer to the operation over discrete sets, and IoU_{Loc} to refer to the operation over spatial regions.

Mathematical Metric Definition. The term *metric* has a strict mathematical definition. For a distance measure (such as HOTA) between two sets (e.g., ground-truth tracks and predicted tracks) to be strictly a *metric* in the mathematical sense, it must satisfy three conditions, (i) identity of indiscernibles, (ii) symmetry, and (iii) subadditivity (the triangle inequality).

Within the computer vision community, the term metric is commonly used for functions which calculate a score by which algorithms can be ranked, without requiring the three conditions above. We use this definition throughout this chapter, and for example, refer to MOTA as an evaluation metric even though it does not meet the last two requirements.

Note that for the purpose of evaluating the MOT task, it is not strictly necessary for metrics to be symmetric or subadditive. However, as we show in Sec. 4.7 these are both useful properties to have. Of the commonly used metrics, HOTA is the only one to have these properties and thus technically be mathematically a *metric*.

4.4 Overview of Previous Metrics

In this section we provide a brief overview of the calculation of the three metrics which are currently used by MOT benchmarks (see Sec. 4.2). Our HOTA metric builds upon many of the ideas of previous metrics (especially MOTA), and as such it is important for the reader to have an overview of how they work. An analysis comparing the properties and deficits of all of the methods is presented in Sec. 4.9, and Tab. 4.1 gives an overview of the different design decisions between the metrics.

4.4.1 CLEARMOT: MOTA and MOTP

Matching Predictions and Ground-Truth. In MOTA (Multi-Object Tracking Accuracy), matching is done at a detection level. A bijective (one-to-one) mapping is constructed between prDets and gtDets in each frame. Any prDets and gtDets that are matched (correct predictions) become true positives (TPs). Any remaining prDets that are not matched (extra predictions) become false positives (FPs). Any gtDets that are not matched (missing predictions) become false negative (FNs). prDets and gtDets can only be matched if they are adequately spatially similar. MOTA thus requires the definition of a similarity score, S , between detections (e.g. IoU_{Loc} for 2D bounding boxes), and the definition of a threshold, α , such that detections are only allowed to match when $S \geq \alpha$. In practice, multiple matches could occur, the actual matching is performed such that the final MOTA and MOTP scores are optimised (see below).

Measuring Association. In MOTA, association is measured with the concept of an Identity Switch (IDSW). An IDSW occurs when a tracker wrongfully swaps object identities or when a track was lost and reinitialised with a different identity. Formally, an IDSW is a TP which has a prID that is different from the prID of the previous TP (that has the same gtID). IDSWs only measure association errors compared to the single previous TP, and do not count errors where the same prID swaps to a different gtID (ID Transfer).

Scoring Function. MOTA measures three types of tracking errors. The detection errors of FNs and FPs, as well as the association error of IDSW. The final MOTA score is calculated by summing up the total number of these errors, dividing by the number of gtDets, and subtracting from one.

$$\text{MOTA} = 1 - \frac{|\text{FN}| + |\text{FP}| + |\text{IDSW}|}{|\text{gtDet}|} \quad (4.2)$$

MOTP.. Note that MOTA does not include a measure of localisation error. Instead the CLEAR MOT metrics define a secondary metric, MOTP (Multi-Object Tracking Precision), which solely measures localisation accuracy. It is simply the average similarity score, $\mathcal{S}$, over the set of TPs.

$$\text{MOTP} = \frac{1}{|\text{TP}|} \sum_{\text{TP}} \mathcal{S} \quad (4.3)$$

Matching to Optimise MOTA and MOTP.. The matching of prDets to gtDets is performed so that the final MOTA and MOTP scores are maximised. This is implemented, in each new frame, by first fixing matches in the current frame which have $\mathcal{S} \geq \alpha$ and do not cause an IDSW. For the remaining potential matches the Hungarian algorithm is run to select the set of matches that as a first objective maximises the number of TPs, and as a secondary objective maximises the mean of $\mathcal{S}$ across the set of TPs.

4.4.2 The Identification Metrics: IDF1

Matching Predictions and Ground-Truth. IDF1 calculates a bijective (one-to-one) mapping between the sets of gtTrajs and prTrajs (unlike MOTA which matches at a detection level). This defines new types of detection matches. IDTPs (identity true positives) are matches on the overlapping part (where $\mathcal{S} \geq \alpha$) of trajectories that are matched together. IDFNs (identity false negatives) and IDFPs (identity false positives) are the remaining gtDets and prDets respectively, from both non-overlapping sections of matched trajectories, and from the remaining trajectories that are not matched.

Scoring Function. The ID-Recall, ID-Precision and IDF1 scores are calculated as follows:

$$\text{ID-Recall} = \frac{|\text{IDTP}|}{|\text{IDTP}| + |\text{IDFN}|} \quad (4.4)$$

$$\text{ID-Precision} = \frac{|\text{IDTP}|}{|\text{IDTP}| + |\text{IDFP}|} \quad (4.5)$$

$$\text{IDF1} = \frac{|\text{IDTP}|}{|\text{IDTP}| + 0.5 |\text{IDFN}| + 0.5 |\text{IDFP}|} \quad (4.6)$$

Matching to Optimise IDF1. The matching is performed such that IDF1 is optimised. This is performed by enumerating the number of IDFPs and IDFNs that would result from each match (non-overlapping sections), and from non-matched trajectories (number of prDet and gtDet in these trajectories, respectively). The Hungarian algorithm is used to select which trajectories to match so that the sum of the number of IDFPs and IDFNs is minimised. Note that the localisation accuracy is not minimised during IDF1 matching unlike in MOTA.

4.4.3 Track-mAP

Matching Predictions and Ground-Truth. Track-mAP (mean average precision) matches predictions and ground-truth at a trajectory level. It requires the definition of a trajectory similarity score, $\mathcal{S}_{\text{tr}}$, between trajectories (in contrast to MOTA and IDF1 which use a detection similarity score, $\mathcal{S}$), as well as a threshold α_{tr} such that trajectories are only allowed to match if $\mathcal{S}_{\text{tr}} \geq \alpha_{\text{tr}}$. A prTraj is matched with a gtTraj if it has the highest confidence score of all prTrajs with $\mathcal{S}_{\text{tr}} \geq \alpha_{\text{tr}}$. If one prTraj should match with more than one gtTraj, it is matched with the one for which it has the highest $\mathcal{S}_{\text{tr}}$, and the other gtTrajs can be matched by the prTraj with the next highest confidence score. We define the matched prTrajs as true positive trajectories (TPTr), and the remaining prTrajs as false positive trajectories (FPTr).

Trajectory Similarity Scores. $\mathcal{S}_{\text{tr}}$ is commonly defined for 2D bounding box tracking in two different ways.

In (Russakovsky et al., 2015; Zhu et al., 2020), the set of TPs is defined as pairs of detections in the trajectories where $\mathcal{S} \geq \alpha$ (with $\mathcal{S}$ being IoU_{Loc} , and α being 0.5). FNs and FPs are the remaining gtDets and prDets respectively. $\mathcal{S}_{\text{tr}}$ is then equal to $|\text{TP}| / (|\text{TP}| + |\text{FN}| + |\text{FP}|)$.

In (Yang et al., 2019; Dave et al., 2020), $\mathcal{S}_{\text{tr}}$ is defined as the sum of the spatial intersection of the boxes across the whole trajectories, divided by the sum of the spatial union of the boxes across the whole trajectories.

Scoring Function. Track-mAP follows the calculation of the average precision metric (Everingham et al., 2010; Lin et al., 2014; Russakovsky et al., 2015) over trajectories (instead of detections as commonly used for evaluating object detection).

PrTrajs are ordered by decreasing confidence score. Let the index of this ordering (starting at one) be n . Let the number of TPTrs in this list up to index n be $|\text{TPTr}|_n$. For each value of n , the precision (Pr_n) and recall (Re_n) can be calculated as:

$$\text{Pr}_n = \frac{|\text{TPTr}|_n}{n} \quad (4.7)$$

$$\text{Re}_n = \frac{|\text{TPTr}|_n}{|\text{gtTraj}|} \quad (4.8)$$

The precision values are interpolated (InterpPr_n) so that they are monotonically decreasing.

$$\text{InterpPr}_n = \max_{m \geq n} (\text{Pr}_m) \quad (4.9)$$

The Track mAP score is then the integral under the interpolated precision-recall curve created by plotting InterpPr_n against Re_n for all values of n . This integral is approximated by averaging over a number of fixed recall values.

Threshold and Class Averaging. Track-mAP is sometimes calculated at a fixed value of α_{tr} (Dave et al., 2020), and sometimes averaged over a range different α_{tr} values (Yang et al., 2019).

When multiple classes are to be tracked, Track-mAP is usually calculated per class separately and then the final score is averaged over the classes.

4.5 The HOTA Evaluation Metric

The main contribution of this chapter is a novel evaluation metric for evaluating Multi-Object Tracking (MOT) performance. We term this evaluation metric HOTA (Higher Order Tracking Accuracy). HOTA builds upon the previously used MOTA metric (Multi-Object Tracking Accuracy) (Bernardin & Stiefelhagen, 2008), while addressing many of its deficits.

HOTA is designed to: (i) provide a single score for tracker evaluation which fairly combines all different aspects of tracking evaluation, (ii) evaluate long-term higher-order tracking association, and finally, (iii) decompose into sub-metrics which allow analysis of the different components of tracker’s performance.

In this section, we provide a definition of the HOTA metric. In Sec. 4.6 we show how HOTA can be decomposed into a set of sub-metrics which can be used to analyse

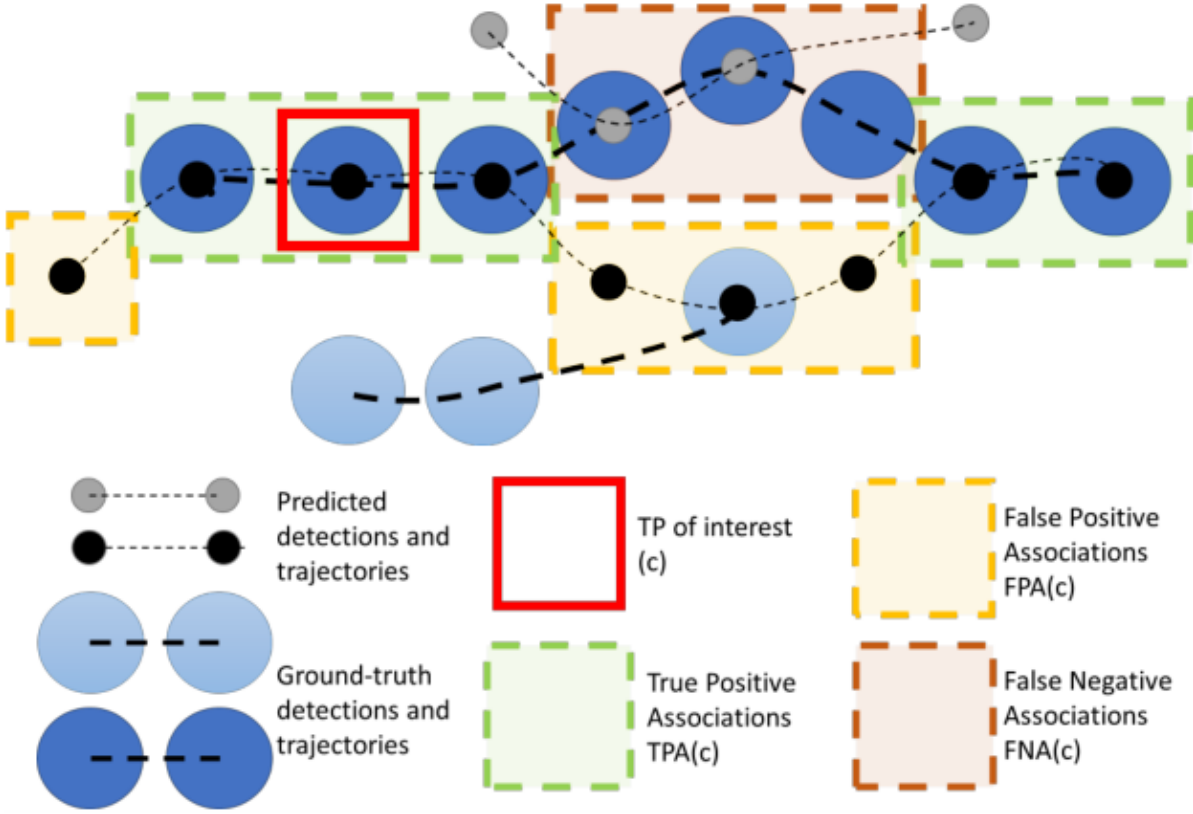


Figure 4.2: A visual explanation of the concepts of TPA, FPA and FNA. The different TPAs, FPAs and FNAs are highlighted for the TP of interest c . The TPAs (green) for c (red) are the matches which have the same prID and the same gtID. The FPAs have the same prID but either a different or no gtID. The FNAs have the same gtID but either a different or no prID. In the diagram, c has five TPAs, four FPAs and three FNAs. Conceptually these concepts are trying to answer the question: *For the matched TP c , how accurate is the alignment between the gtTraj for this TP (large dark blue circles) and the prTraj for this TP (small black circles).*

different aspects of tracking performance. In Sec. 4.7 we analyse different properties of HOTA, and examine the design decisions inherent to the HOTA formulation. In Sec. 4.9 we present a comparison of HOTA to MOTA, IDF1 and Track-mAP, and show how HOTA addresses many of the deficits of previous metrics.

Matching Predictions and Ground-Truth. In HOTA, matching occurs at a detection level (similar to MOTA). A true positive (TP) is a pair consisting of a gtDet and a prDet, for which the localisation similarity $\mathcal{S}$ is greater than or equal to the threshold α . A false negative (FN) is a gtDet that is not matched to any prDet. A false positive (FP) is a prDet that is not matched to any gtDet. The matching between gtDets and prDets is

bijjective (one-to-one) in each frame. Multiple different combinations of matches could occur, the actual matching is performed to maximise the final HOTA score (see below).

Measuring Association. The concepts of TPs, FNs and FPs are commonly used to measure detection accuracy. In order to evaluate the success of association in a similar way, we propose the novel concepts of TPAs (True Positive Associations), FNAs (False Negative Associations) and FPAs (False Positive Associations), which are defined for each TP. For a given TP, c , the set of TPAs is the set of TPs which have both the same gtID and the same prID as c :

$$\begin{aligned} \text{TPA}(c) &= \{k\}, \\ k &\in \{\text{TP} \mid \text{prID}(k) = \text{prID}(c) \wedge \text{gtID}(k) = \text{gtID}(c)\} \end{aligned} \quad (4.10)$$

For a given TP, c , the set of FNAs is the set of gtDets with the same gtID as c , but that were either assigned a different prID as c , or no prID if they were missed:

$$\begin{aligned} \text{FNA}(c) &= \{k\}, \\ k &\in \{\text{TP} \mid \text{prID}(k) \neq \text{prID}(c) \wedge \text{gtID}(k) = \text{gtID}(c)\} \\ &\quad \cup \{\text{FN} \mid \text{gtID}(k) = \text{gtID}(c)\} \end{aligned} \quad (4.11)$$

Finally, for a given TP, c , the set of FPAs is the set of prDets with the same prID as c , but that were either assigned a different gtID as c , or no gtID if they did not actually correspond to an object:

$$\begin{aligned} \text{FPA}(c) &= \{k\}, \\ k &\in \{\text{TP} \mid \text{prID}(k) = \text{prID}(c) \wedge \text{gtID}(k) \neq \text{gtID}(c)\} \\ &\quad \cup \{\text{FP} \mid \text{prID}(k) = \text{prID}(c)\}. \end{aligned} \quad (4.12)$$

A visual example explaining the concept of TPAs, FNAs and FPAs is shown in Fig. 4.2.

Note that although TPAs, FPAs, and FNAs are measured between pairs of detections, these measures can be easily and efficiently calculated by counting the number of matches per each prID-gtID pair, and there is no need to explicitly iterate over all pairs of detections.

Scoring Function. Now we have defined concepts used to measure successes and errors in detection (TPs, FPs, FNs) and association (TPAs, FPAs, FNAs), we can define the HOTA_α score for a particular localisation threshold α :

$$\text{HOTA}_\alpha = \sqrt{\frac{\sum_{c \in \{\text{TP}\}} \mathcal{A}(c)}{|\text{TP}| + |\text{FN}| + |\text{FP}|}} \quad (4.13)$$

$$\mathcal{A}(c) = \frac{|\text{TPA}(c)|}{|\text{TPA}(c)| + |\text{FNA}(c)| + |\text{FPA}(c)|} \quad (4.14)$$

We call this a “double Jaccard” formulation, where a typical Jaccard metric is used over detection concepts of TPs/FPs/FNs with each of the TPs in the numerator being weighted by an association score $\mathcal{A}$ for that TP, which is equal to another Jaccard metric, but this time over the association concepts of TPAs/FNAs/FPA.

$\mathcal{A}$ measures the alignment between the gtTraj and prTraj which are matched at the TP c . This alignment is calculated using the same formulation (Jaccard) as is used to measure the alignment between the whole set of all gtTrajs and all prTrajs for detection, but in this case only over the subset of trajectories that are matched at a TP.

Note that concepts such as $\mathcal{A}$, TP, FN, FP, TPA, FNA and FPA, are all calculated for a particular value of α . However, the α subscript is omitted for clarity. We will continue to omit this α throughout the chapter except where it is needed for understanding.

Matching to Optimise HOTA.. Like in MOTA (and IDF1) the matching occurs in HOTA to maximise the final HOTA score. The Hungarian algorithm is run to select the set of matches, such that as a first objective the number of TPs is maximised, as a secondary objective the mean of the association scores $\mathcal{A}$ across the set of TPs is maximised, and as a third objective the mean of the localisation similarity $\mathcal{S}$ across the set of TPs is maximised. This is implemented with the following scoring for potential matches, MS, between each gtDet i and each prDet j .

$$\text{MS}(i, j) = \begin{cases} \frac{1}{\epsilon} + \mathcal{A}_{\max}(i, j) + \epsilon \mathcal{S}(i, j) & \text{if } \mathcal{S}(i, j) \geq \alpha \end{cases} \quad (4.15)$$

where ϵ is small number such that the components have different magnitudes. $\mathcal{A}_{\max}$ is the maximum $\mathcal{A}$ score if detections are not required to be bijectively matched, e.g. if each prDet and each gtDet are allowed to match with multiple others. $\mathcal{A}_{\max}$ is optimised as a proxy for $\mathcal{A}$. This is because $\mathcal{A}$ depends upon which matches are selected, and thus cannot be optimised using a linear assignment formulation. This approximation is valid, because $\mathcal{A}$ approaches $\mathcal{A}_{\max}$ for the optimal assignment.

Integrating over Localisation Thresholds. The formulation for HOTA_α in Eq. 4.13 accounts for both detection and association accuracy, but does not take into account localisation accuracy. In order for HOTA to measure localisation, the final HOTA score is the integral (area under the curve) of the HOTA score across the valid range of α values between 0 and 1. This is approximated by evaluating HOTA at a number of

different distinct α values (0.05 to 0.95 in 0.05 intervals) and averaging over these values. For each α value the matching between gtDets and prDets is performed separately.

$$\text{HOTA} = \int_0^1 \text{HOTA}_\alpha d\alpha \approx \frac{1}{19} \sum_{\alpha \in \{\frac{0.05}{0.9}, \frac{0.1}{0.95}, \dots\}} \text{HOTA}_\alpha \quad (4.16)$$

HOTA in One Sentence. If HOTA were to be described in one sentence it would be:
HOTA measures how well the trajectories of matching detections align, and averages this over all matching detections, while also penalising detections that do not match.

4.6 Decomposing HOTA into Different Error Types.

A set of evaluation metrics has two main purposes. The first purpose is to enable simple comparison between methods to determine which perform better than others. For this purpose it is important that there exists a single metric by which methods can be ranked and compared, for this we propose the HOTA metric (see Sec 4.5). The second purpose of evaluation metrics is to enable the analysis and understanding of the different types of errors that algorithms are making, in order to understand how algorithms can be improved, or where they are likely to fail when used. In this section, we show that the HOTA metric naturally decomposes into a family of sub-metrics which are able to separately measure all of the different aspects of tracking. Fig. 4.3 shows each of these sub-metrics and the relations between them.

HOTA solves the long-held debate in the tracking community (Manohar et al., 2006; Milan et al., 2013) about whether it is better to have a single evaluation metric or multiple different metrics. HOTA simultaneously gives users the benefits of both options, a single metric for ranking trackers, and the decomposition into sub-metrics for understanding different aspects of tracking behaviour.

Taxonomy of Error Types. We classify potential tracking errors into three categories: detection errors, association errors and localisation errors. Detection errors can be further categorised into errors of detection recall (measured by FNs) and detection precision (measured by FPs). Association errors can be further categorised into errors of association recall (measured by FNAs) and association precision (measured by FPAs).

Detection recall errors occur when trackers fail to predict detections that are in the ground-truth (misses). Detection precision errors occur when trackers predict extra detections that do not exist in the ground-truth. Association recall errors occur when trackers assign two different prIDs to the same gtTraj. Association precision errors occur when trackers assign the same prID to two different gtTrajs. Localisation errors occur when prDets are not perfectly spatially aligned with gtDets.

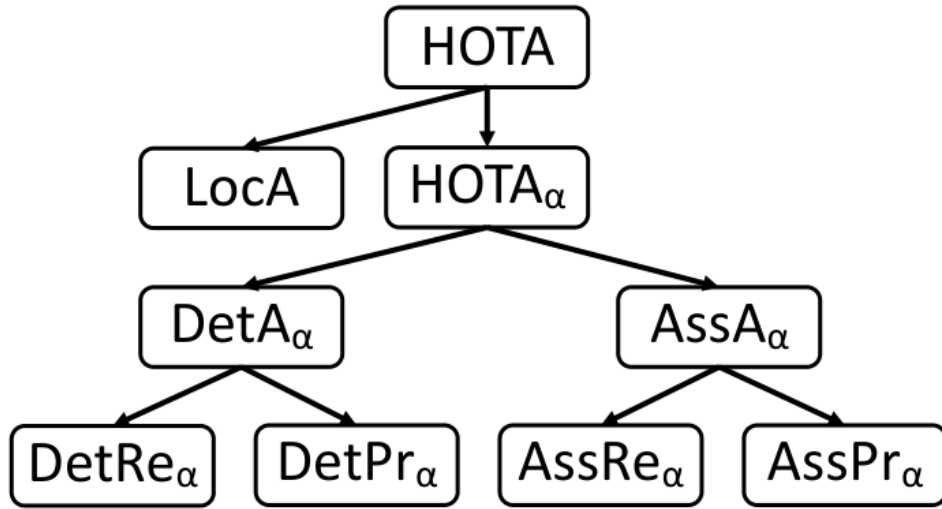


Figure 4.3: Diagrammatic representation of how HOTA can be decomposed into separate sub-metrics which are able to differentiate between different types of tracking errors.

Our taxonomy aligns with previous work to classify different tracking errors (Leichter & Krupka, 2013), which defines the five basic error types as: false negatives, false positives, fragmentations, mergers and deviations. These are equivalent to detection recall, detection precision, association recall, association precision, and localisation, respectively.

Leichter and Krupka (Leichter & Krupka, 2013) argue that any set of tracking metrics must be both error type differentiable and monotonic with respect to these five basic error types. HOTA meets both criteria as it naturally decomposes into separate sub-metrics measuring each basic error type, and is designed to ensure monotonicity (see Sec. 4.7).

Metrics for Different Tracking Scenarios with Different Requirements. The decomposition of HOTA into different sub-metrics has the further advantage that it enables users to select algorithms or tune algorithms’ hyper-parameters based on the nuances of their particular use-case.

For example, in human motion analysis, crowd analysis or sports analysis (Milan et al., 2013; Leal-Taixé et al., 2017) it may be far more important to predict correct, identity preserving trajectories (association recall and precision), than to find all present objects (detection recall). Whereas for a driving assistance system, it is crucial to detect every pedestrian to avoid collision (detection recall) while also not predicting objects that are not present to avoid unnecessary evasive action (detection precision). However, correctly associating these detections over time may be less crucial (association recall and precision). In surveillance scenarios, it is typically more important to ensure all objects are found (detection recall), whereas extra detections can easily be ignored

by human observers (detection precision). For short-term future motion prediction it is important to have accurate trajectories of recent object motion, without mixing trajectories of multiple objects (association precision), in order to extrapolate to future motion. Whereas it does not matter if trajectories are not correctly merged into long-term consistent tracks (association recall). With HOTA, different aspects of tracking can easily be analysed and optimised for, which was not as easily or intuitively possible with previous metrics.

Measuring Localisation. HOTA is calculated at a number of different localisation thresholds α , and the final HOTA score is the average of the HOTA_α scores calculated at each threshold. This formulation ensures that the final HOTA score takes the actual localisation accuracy into account.

A localisation accuracy score (LocA) can be measured separately from other aspects of tracking, as follows:

$$\text{LocA} = \int_0^1 \frac{1}{|\text{TP}_\alpha|} \sum_{c \in \{\text{TP}_\alpha\}} \mathcal{S}(c) d\alpha \quad (4.17)$$

Where $\mathcal{S}(c)$ is the spatial similarity score between the prDet and gtDet which make up the TP c . This is similar to MOTP (Bernardin & Stiefelhagen, 2008), however it is evaluated over multiple localisation thresholds α , similarly to how HOTA is calculated.

Separating Detection and Association. HOTA can be naturally decomposed into a separate detection accuracy score (DetA) and an association accuracy score (AssA) as follows:

$$\text{DetA}_\alpha = \frac{|\text{TP}|}{|\text{TP}| + |\text{FN}| + |\text{FP}|} \quad (4.18)$$

$$\text{AssA}_\alpha = \frac{1}{|\text{TP}|} \sum_{c \in \{\text{TP}\}} \mathcal{A}(c) \quad (4.19)$$

$$\mathcal{A}(c) = \frac{|\text{TPA}(c)|}{|\text{TPA}(c)| + |\text{FNA}(c)| + |\text{FPA}(c)|} \quad (4.20)$$

$$\begin{aligned} \text{HOTA}_\alpha &= \sqrt{\frac{\sum_{c \in \{\text{TP}\}} \mathcal{A}(c)}{|\text{TP}| + |\text{FN}| + |\text{FP}|}} \\ &= \sqrt{\text{DetA}_\alpha \cdot \text{AssA}_\alpha} \end{aligned} \quad (4.21)$$

We see that HOTA is equal to the geometric mean of a detection score and an association score. This formulation ensures that both detection and association are

evenly balanced, unlike many other tracking metrics, and that the final score is somewhere between the two.

It also ensures that both the detection score and association score have the same structure. Both are calculated using a Jaccard index formulation, and both are calculated to ensure that each detection contributes equally to the final score.

The detection score is simply the standard Jaccard index, which is commonly used for evaluating detection. The association score is a sum of Jaccard indices over the TPs, where each is equal to a standard Jaccard index evaluated only between the trajectories which are a part of that TP.

In practice (see Sec. 4.10) this often results in trackers having DetA and AssA scores that are quantitatively similar.

Final DetA and AssA scores can be calculated by integrating over the range of α values in the same way as done for the HOTA score in Eq. 4.16. Note that DetA and AssA should only be combined into HOTA before integrating over a range of α values, not afterwards.

Separating Recall and Precision. HOTA is further decomposable in that each of the detection and association components can be simply decomposed into a recall and precision component.

Detection Recall/Precision. The detection recall/precision are defined as follows:

$$\text{DetRe}_\alpha = \frac{|\text{TP}|}{|\text{TP}| + |\text{FN}|} \quad (4.22)$$

$$\text{DetPr}_\alpha = \frac{|\text{TP}|}{|\text{TP}| + |\text{FP}|} \quad (4.23)$$

$$\text{DetA}_\alpha = \frac{\text{DetRe}_\alpha \cdot \text{DetPr}_\alpha}{\text{DetRe}_\alpha + \text{DetPr}_\alpha - \text{DetRe}_\alpha \cdot \text{DetPr}_\alpha} \quad (4.24)$$

These are equivalent to the concepts commonly used in the field of object detection (Everingham et al., 2010). Detection recall is the percentage of ground-truth detections that have been correctly predicted, and detection precision is the percentage of detection predictions made which are correct. DetRe and DetPr are easily combined into DetA. Final DetRe and DetPr scores can be calculated by integrating over the range of α values in the same way as done for the HOTA score in Eq. 4.16.

Association Recall/Precision. The association recall/precision are defined as follows:

$$\text{AssRe}_\alpha = \frac{1}{|\text{TP}|} \sum_{c \in \{\text{TP}\}} \frac{|\text{TPA}(c)|}{|\text{TPA}(c)| + |\text{FNA}(c)|} \quad (4.25)$$

$$\text{AssPr}_\alpha = \frac{1}{|\text{TP}|} \sum_{c \in \{\text{TP}\}} \frac{|\text{TPA}(c)|}{|\text{TPA}(c)| + |\text{FPA}(c)|} \quad (4.26)$$

$$\text{AssA}_\alpha = \frac{\text{AssRe}_\alpha \cdot \text{AssPr}_\alpha}{\text{AssRe}_\alpha + \text{AssPr}_\alpha - \text{AssRe}_\alpha \cdot \text{AssPr}_\alpha} \quad (4.27)$$

Unlike the detection equivalent, these are novel concepts. Association recall measures how well predicted trajectories cover ground-truth trajectories. E.g. a low AssRe will result when a tracker splits an object up into multiple predicted tracks. Association precision measures how well predicted trajectories keep to tracking the same ground-truth trajectories. E.g. a low AssPr will result if a predicted track extends over multiple objects.

AssRe and AssPr are easily combined into AssA. The introduction of association precision and recall are very powerful tools for measuring different aspects of MOT performance and are a natural extension to the similar widely used detection concepts. Final AssRe and AssPr scores can be calculated by integrating over the range of α values in the same way as done for the HOTA score in Eq. 4.16.

4.7 Analysing the Design Space of HOTA

In this section, we analyse a number of different design choices of the HOTA algorithm, and the effect of these choices on the properties of the evaluation metric.

Higher-Order vs First-Order Association. In HOTA (and in MOTA) the concept of an association is measured for each detection. The association score in HOTA, and the number of IDSWs in MOTA seek to answer the question “how well is this detection associated throughout time?”. In MOTA an IDSW measures this association only one time-step back into the past. E.g. whether this detection has the correct association compared to the previous detection. Since association is measured only over one step, we term this *first-order association*. A metric which considers associations over two time-steps could be called second order association. In contrast, HOTA measures association over all frames of a video. We term this concept *higher-order association* and name our HOTA metric after it. This property allows HOTA to measure long-term association, which is lacking from the MOTA metric.

Higher-Order vs First-Order Matching. Just as one of the main conceptual differences between HOTA and MOTA is first-order vs higher-order association between detections,

one of the main conceptual differences between HOTA and IDF1 can be thought of as first-order vs higher-order matching between trajectories.

In IDF1, each trajectory is matched only with a single other trajectory and scored by how well it aligns with this single trajectory. We call this *first-order matching*. This is enforced by a unique bijective matching between prTrajs and gtTrajs. HOTA in contrast, is able to measure how well each trajectory matches with all possible matching trajectories, which we term *higher-order matching*. This is done by performing matching at a detection level, which allows each trajectory to match to different trajectories in each time-step, and then scoring the alignment between each of these matching trajectories for each matching detection.

Thus HOTA can be thought of as being *higher-order* in terms of both association and matching.

Detection vs Trajectory Matching. Both HOTA and MOTA create matches between sets of detections, whereas other metrics like IDF1 and Track-mAP directly match whole trajectories with one another. Matching detections has the advantage over matching trajectories that all of the possible trajectory matches can be measured by the metric simultaneously. In IDF1, if a gtTraj is split between multiple prTrajs, only the best matching trajectories are considered correct, while all the remaining trajectories are considered incorrect. This causes the problem that the association accuracy of non-matched trajectories are ignored, no matter how well they are associated it will not affect the score. This also has the disadvantage that the score actually decreases as detection accuracy increases for non-matched segments. This is because such segments are considered negatives and decrease the score. Matching at a detection level instead of a trajectory level is required in order to ensure that the association accuracy of all segments contributes to the final score, and that the metric monotonically increases as detection improves.

Jaccard vs F1 vs MODA.. In HOTA we use the Jaccard index to measure both detection and association accuracy. We compare this formulation with two possible alternatives, the F1 score and the MODA score.

$$\text{Jaccard} = \frac{|\text{TP}|}{|\text{TP}| + |\text{FN}| + |\text{FP}|} \quad (4.28)$$

$$\text{F1} = \frac{|\text{TP}|}{|\text{TP}| + 0.5|\text{FN}| + 0.5|\text{FP}|} \quad (4.29)$$

$$\text{MODA} = \frac{|\text{TP}| - |\text{FP}|}{|\text{TP}| + |\text{FN}|} \quad (4.30)$$

The F1 score (also called the Dice coefficient) is the harmonic mean of recall and precision and is used in IDF1 as well as other metrics such as PQ for panoptic segmentation (Kirillov et al., 2019). The MODA score (Multi-Object Detection Accuracy) is the MOTA

score without considering IDSWs and is often used for measuring detection accuracy in video.

Of the three, the Jaccard formulation is the only one that meets all three requirements to mathematically be a *metric*, and the only one that obeys the triangle inequality (MODA also is not symmetric). Note that when using the Jaccard formulation the entire HOTA metric also obeys the triangle inequality, and is mathematically a *metric*, because the mean of metrics over non-empty finite subsets of a metric space is also a metric (Fujita, 2013).

Fig. 4.4 shows a simple tracking example designed to show the monotonicity of different formulations. The HOTA score is evaluated using the three different formulations for both detection and association scores. If the evaluation measure is monotone, the tracking results in (A) should always be higher than (B) because (A) contains more correct detections. This should also be true when swapping which set is the ground-truth and which is the prediction. As can be seen of the three formulations, Jaccard is the only one to exhibit this monotone property. The F1 formulation scores B higher when the track labeled (1) is adequately long. The MODA formulation is non-symmetric. It acts the same as Jaccard in the absence of TPs and TPAs, but exhibits very undesirable behaviour when the ground-truth and predictions are switched. Both the monotone and symmetric properties result in the Jaccard formulation being preferable to the other two.

Including vs Excluding Detection Errors in the Association Score. In HOTA, the association score for a TP is the Jaccard index score between the gtTraj and the prTraj that have the same gtID and prID as the TP, respectively. These trajectories could include FNs and FPs which are not matched. This can be seen in the definitions of TPA and TNA in Sec. 4.5. A potential alternative formulation would only calculate this Jaccard index over TPs such that any FNs or FPs in these trajectories are ignored, and do not count toward the count of FNAs and FPAs. This formulation may seem advantageous in that association is now calculated only over TPs, and detection errors would no longer decrease the association score. However, as seen in the fourth panel of Fig. 4.4 this results in non-monotonic results where adding in correct detections decreases the overall score, as the AssA decreases faster than the DetA increases. By including all detections of matching trajectories in the association score calculation, we ensure that DetA and AssA are perfectly balanced such that an improvement in either one cannot result in a larger decrease in the other, thus ensuring the monotonicity of the metric.

Note that in HOTA, the presence of these FNs and FPs in the association score does not influence the *error type differentiability* of the metric. The AssA still measures only association and the DetA still measures only detection. These terms in the association score correspond to measuring associations to unmatched detections.

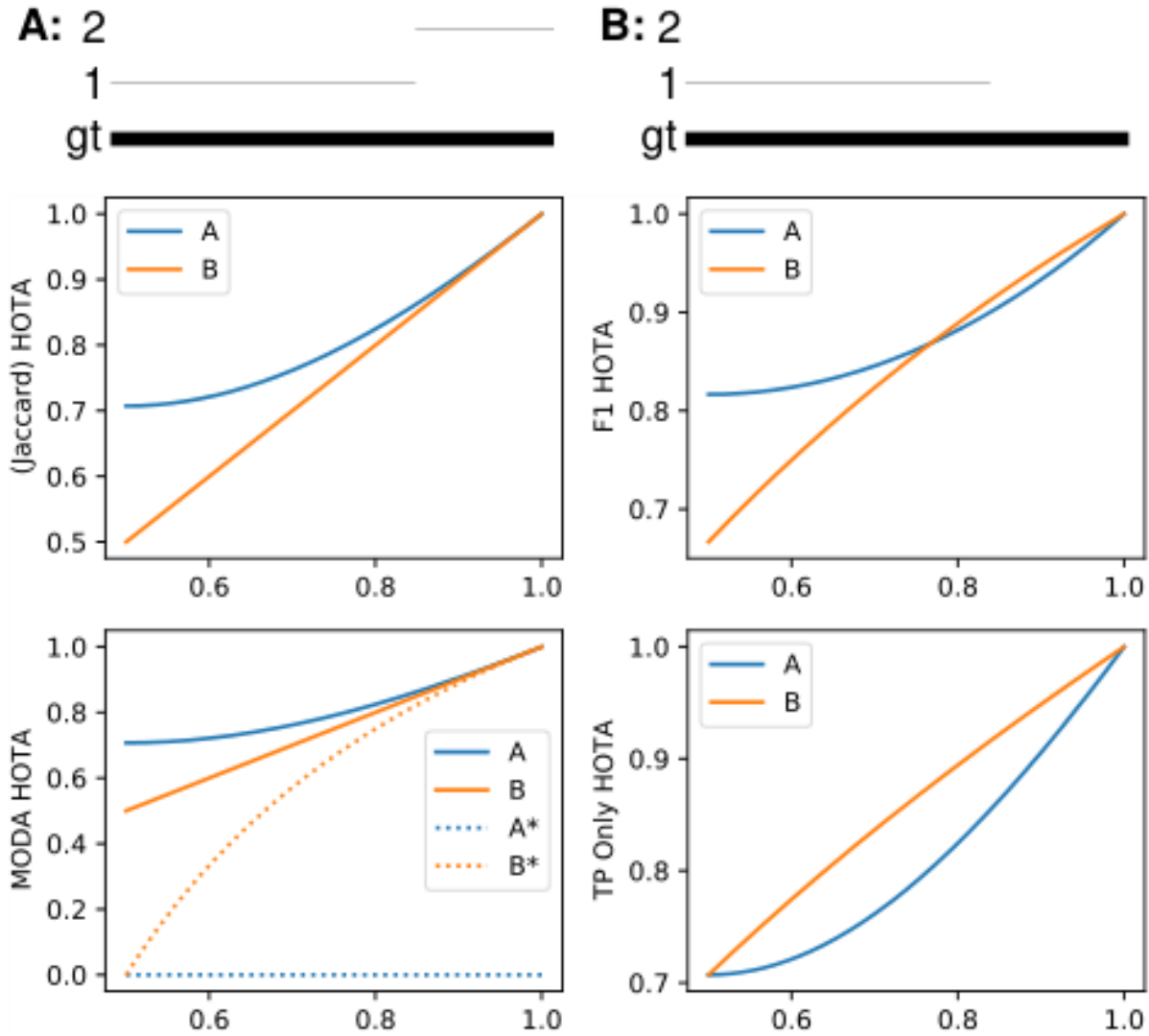


Figure 4.4: A simple tracking example showing why the Jaccard formulation for HOTA is preferable to a number of others such as F1, MODA and a formulation which excludes FNs and FPs from the association score (see text). For tracking a single object present in all frames (bold line), two tracking results, A and B (thin lines) are presented. The x-axis of the plots is the ratio of the $\text{len}(\text{traj } 1) / \text{len}(\text{gt})$. The tracking result of A should always have a higher score than B for the metric to be monotonic in detection, over all possible ratios of the $\text{len}(\text{traj } 1) / \text{len}(\text{gt})$, and also if the predictions and ground-truth are swapped. This is only valid for Jaccard based HOTA. Note that the MODA formulation is non-symmetric so the results when swapping the ground-truth and tracks are shown as dashed lines and labeled with an asterisk. The other formulations are symmetric.

With vs Without the Square Root. The HOTA formulation contains a square root operation after the double Jaccard formulation. This square root has three effects. The first is that it increases the magnitude and spread of trackers' scores. While the magnitude of the scores is in itself not important, it is nice to see that both the magnitude and spread of HOTA scores is in the same ball-park range as previous metrics (see Sec. 4.10 Fig. 4.13). This means that researchers' current intuitive understanding of how good certain scores are, still roughly holds.

The second effect of the square root is its interpretation as the geometric mean of a detection score and an association score. This is natural and intuitive as we wish for HOTA to evenly balance both detection and association, thus having HOTA as the geometric mean of these two scores is a good choice. The geometric mean has the advantage over other formulations such as an arithmetic mean that the score approaches 0 as either of the two sub-scores approach zero. Thus when a tracker completely fails in either detection or association, a very low score will effectively represent this.

The third effect is that it accounts for double counting of similar error types. As discussed above, the association score also includes FP and FN detection errors. The use of the square root prevents double counting of these errors. This can be illustrated by a simple example. In cases where there is only one gtTraj and one prTraj, such as in single object tracking (SOT), the HOTA score reduces to the following equation:

$$\begin{aligned}
 \text{HOTA}_\alpha\{\text{SOT}\} &= \sqrt{\text{DetA} \cdot \text{AssA}} \\
 &= \sqrt{\frac{|\text{TP}|}{|\text{TP}| + |\text{FN}| + |\text{FP}|} \cdot \frac{1}{|\text{TP}|} \sum_{|\text{TP}|} \frac{|\text{TP}|}{|\text{TP}| + |\text{FN}| + |\text{FP}|}} \\
 &= \frac{|\text{TP}|}{|\text{TP}| + |\text{FN}| + |\text{FP}|}
 \end{aligned} \tag{4.31}$$

In the single object tracking case, the association score and the detection score are the same. This makes sense because the association score can be thought of as the average detection score between matched trajectories, and where there is only one possible matching trajectory these scores are the same. The square root thus cancels out these same errors being counted twice, and results in an intuitive metric for SOT, which is just a Jaccard metric measuring the ratio of correct to incorrect predictions.

With vs Without the Detection Accuracy. As we have seen above, the AssA takes into account FP and FN detection errors for matched trajectories. A natural question to ask then, is whether we even need the DetA, or if the AssA is adequate by itself. The DetA is critical for two reasons, the first is in accounting for non-matched trajectories. If we have gtTrajs or prTraj that are not matched at all, then these FNs and FPs will not

at all be taken into account in the AssA. The DetA is needed to penalise these errors. The second reason is that the AssA by itself is non-monotonic. Since it is the average association over detections, if we add in one extra matching detection (TP), if this TP has a low association score, then the overall AssA will decrease. By including the DetA in the HOTA score, the DetA and AssA are perfectly balanced such that adding any correct match (TP) will always increase the overall score and thus HOTA is monotonic.

Averaging over Detections vs Averaging over Associations. In order to calculate the AssA we average the association scores between matching trajectories over matching detections. This is a natural formulation because it ensures that each detection contributes equally to the AssA, and it results in the intuitive understanding that the contribution for each detection is weighted by how accurately that detection is associated across time.

An alternative formulation would be to calculate AssA by averaging over all possible associations rather than averaging over detections. This would result in a quadratic dependence on the number of TPs such that longer matching trajectories would contribute quadratically more than shorter matching trajectories, which is undesirable. It would also lose the interpretation that the contribution of each TP is weighted by its association accuracy, as well as making HOTA non-monotonic.

Final Tracks vs Potential Tracks with Confidence Scores. Taking into account the confidence of predictions has both advantages and disadvantages for metrics. One of the main disadvantages is that your metric is no longer evaluating an actual final tracking result, but rather a selection of potential tracking results which are ranked by how likely each one is. Such an approach makes sense when the ground-truth is inherently ambiguous (such as trajectory forecasting) as we cannot expect algorithms to predict the correct results and as such it is only fair to allow multiple ranked predictions to be evaluated. It also makes sense when the task is too difficult for current algorithms to accurately predict the correct ground-truth (such as monocular 3D detection and tracking), and again it is fair to allow algorithms to predict multiple ranked results.

However, as algorithms become better at a task, it is better to evaluate an algorithm's ability to actually predict the correct ground-truth values. This also has other benefits such as enabling constraints between detection representations such as commonly used in segmentation mask tracking (Voigtlaender et al., 2019b) where segmentation masks are not allowed to overlap.

For these reasons, HOTA is designed to operate on final tracking results. We also present an extension to HOTA in Sec. 4.8 in which we present a confidence ranked version of HOTA, which reduces to the default HOTA when taking a fixed set of detections above a certain confidence threshold.

Drawbacks of HOTA.. HOTA has two main potential drawbacks, which could make using it less than ideal in some situations. The first is that it may not be ideal for

evaluating online tracking. This is because association is measured over the entire video, and the association score for each TP depends on how well it is associated in the future, which is not a desirable feature for online evaluation.

The second potential drawback is that it does not take fragmentation of tracking results into account (see Fig. 4.5). This is by design, as we wish for the metric to measure long-term global tracking. However, for applications where this is important, this could be a drawback of HOTA.

In Sec. 4.8 we present both an online version of HOTA and a fragmentation-aware version of HOTA, which address both of these issues.

4.8 HOTA Extensions

In this section we provide a number of different extensions to HOTA for use in different tracking scenarios.

Variety of MOT scenarios. HOTA is designed for a variety of MOT scenarios, from 2D box tracking, segmentation mask tracking, 3D tracking, human pose tracking or point tracking or beyond. HOTA can easily be adapted to any such representation, all that is required is measure of similarity $\mathcal{S}$ between objects in whichever representation is chosen (see Sec. 4.3).

Multi-Camera HOTA.. HOTA also extends trivially to evaluating Multi-Camera MOT. Both the ground-truth and predictions could contain trajectories in multiple different cameras with consistent ids across cameras. HOTA could then be applied without any changes. We recommend the use of HOTA for multi-camera tracking over the currently used IDF1 metric due to the many advantages of HOTA and drawbacks of IDF1 (see Sec. 4.9).

HOTA for Single Object Tracking. As can be seen in Eq. 4.31, HOTA simplifies trivially in the single object tracking (SOT) case to being a Jaccard index (integrated over α values). Currently in SOT most evaluation procedures do not penalise FPs and only evaluate over frames where a gtDet is present. We believe that the HOTA formulation is also perfectly suited for SOT evaluation, both for effectively penalising FPs, as well as for promoting unification between the MOT and SOT communities.

Online HOTA.. By default, HOTA scores association globally over the whole video sequence. Thus the association score for a particular TP depends on how well it is associated both forward and backward in time. For online tracking scenarios (such as autonomous vehicles) this is not ideal evaluation behaviour as the results from the tracker are used online in each time-step (for decision making) and should be evaluated in a similar way.

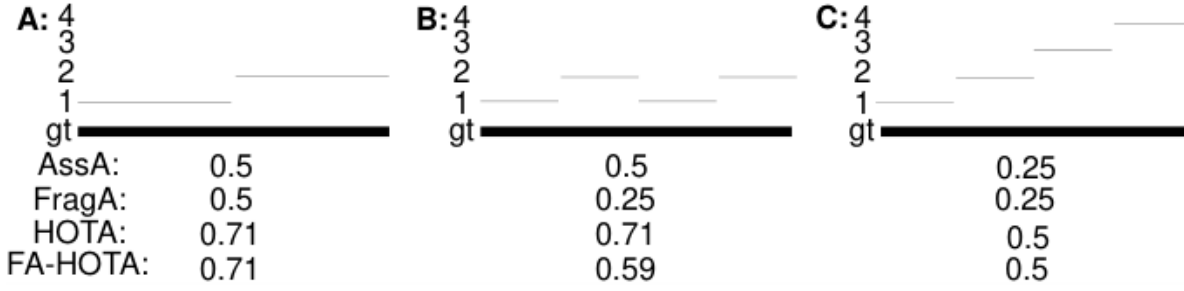


Figure 4.5: An example showing the difference between fragmentation and association. One gtTraj is present in all frames (bold line), and three tracking results, A, B and C (sets of thin lines) are presented. A and B have equal association (global alignment), whereas they have different fragmentation (short-range alignment). B and C have equal fragmentation but different association. HOTA only measures association by design. FA-HOTA measures both association and fragmentation.

Thus we propose a simple extension of HOTA to the online case which we call Online HOTA (OHOTA). OHOTA is calculated in the same way as HOTA but only time-steps up to the current time-step are used for calculating the association accuracy for each TP.

This is a natural way to extend HOTA to online scenarios, and has the further benefit that it can also be used to evaluate online tracking where trackers are able to update previous predictions in each new time-step. In this case, for each new time-step the TPs for this time-step will have their association scores calculated with the most up to date predictions from all previous time-steps.

Fragmentation-Aware HOTA.. HOTA is designed to evaluate global association alignment between gtTrajs and prTrajs. However, in some cases it is important to measure short-range alignment, which we call fragmentation. Fig. 4.5 clearly shows the difference between association and fragmentation.

In most tracking scenarios the default version of HOTA is preferable, however, for when measuring fragmentations is important, we present an extension which we call fragmentation-aware HOTA (FA-HOTA).

$$\text{FA-HOTA}_\alpha = \sqrt{\frac{\sum_{c \in \{\text{TP}\}} \left(\sqrt{\mathcal{A}(c) \cdot \mathcal{F}(c)} \right)}{|\text{TP}| + |\text{FN}| + |\text{FP}|}} \quad (4.32)$$

$$\mathcal{F}(c) = \frac{|\text{FrA}(c)|}{|\text{TPA}(c)| + |\text{FNA}(c)| + |\text{FPA}(c)|} \quad (4.33)$$

where the set of fragment associations of c , $\text{FrA}(c)$, is the subset of $\text{TPA}(c)$ which belongs to the same fragment as c . A fragment is a set of TPAs for which there are no FNAs or FPAs between them.

We compute the geometric mean of fragmentation and association for each TP in order to concurrently measure both short-term fragmentation alignment and long-term association. As the fragmentation score is bounded by the association score, FA-HOTA equals HOTA when no fragmentation occurs. This formulation also allows us to compute the overall fragmentation accuracy FragA:

$$\text{FragA} = \frac{1}{|\text{TP}|} \sum_{c \in \{\text{TP}\}} \text{Frag}(c). \quad (4.34)$$

Importance Weighted HOTA.. As detailed in Sec. 4.6 different tracking applications can assign different importance to different aspects of tracking (detection/association recall/precision). We present an extension of HOTA, Weighted HOTA (W-HOTA) which allows users to apply different weights to each aspect depending on their requirements.

$$\text{W-HOTA}_\alpha = \sqrt{\frac{\sum_{c \in \{\text{TP}\}} \mathcal{A}_w(c)}{|\text{TP}| + w_{\text{FN}}|\text{FN}| + w_{\text{FP}}|\text{FP}|}} \quad (4.35)$$

$$\mathcal{A}_w(c) = \frac{|\text{TPA}(c)|}{|\text{TPA}(c)| + w_{\text{FNA}}|\text{FNA}(c)| + w_{\text{FPA}}|\text{FPA}(c)|} \quad (4.36)$$

Where each of the weightings, w_{FN} , w_{FP} , w_{FNA} , w_{FPA} are values between 0 and 1. When all weightings are 1 we have the original HOTA. When any single weight is 0 that component no longer contributes to the score.

The default weighting provides a balanced weighting between the different components, and should be used for tracking evaluation unless there is a strong reason to use a different weighting for a particular desired outcome.

Classification-Aware HOTA.. Traditionally, there are two ways to deal with evaluation for tracking multiple classes. The first option is to require trackers to assign each object to a class and then evaluate over each class separately before averaging the results over classes. A second option is to ignore the effect of classification all together and simply evaluate all classes together in a class-agnostic way as though they were all the same class.

We propose a third option for dealing with evaluating multiple classes which we call classification-aware HOTA (CA-HOTA). We require that each prediction is assigned a probability that it belongs to each class such that these probabilities sum to one over all classes. CA-HOTA then becomes:

$$\text{CA-HOTA}_\alpha = \sqrt{\frac{\sum_{c \in \{\text{TP}\}} (\mathcal{A}(c) \cdot \mathcal{C}(c))}{|\text{TP}| + |\text{FN}| + |\text{FP}|}} \quad (4.37)$$

where $\mathcal{C}(c)$ is probability that the prDet of c assigned to the class of the gtDet of c . This effectively weights the contribution of each TP by the classification score, in the same way that it is weighted by an association score. In this setting we have to include the classification score in the matching procedure so that the matching still maximises the final score. Eq. 4.15 now becomes:

$$\begin{aligned} \text{MS}(i, j) &= \begin{cases} \frac{1}{\epsilon} + \mathcal{A}_{\max}(i, j) \cdot \mathcal{C}(i, j) + \epsilon \mathcal{S}(i, j) & \text{if } \mathcal{S}(i, j) \geq \alpha \\ 0, & \text{otherwise} \end{cases} \end{aligned} \quad (4.38)$$

Each detection, even those belonging to the same prTraj can have different class probabilities.

We can also compute the overall classification accuracy ClaA, in order to evaluate the success of classification separate from other aspects of tracking.

$$\text{ClaA} = \frac{1}{|\text{TP}|} \sum_{c \in \{\text{TP}\}} \mathcal{C}(c). \quad (4.39)$$

Class-Averaged Classification-Aware HOTA.. We can also calculate a class-averaged classification-aware HOTA (CA²-HOTA), by calculating a score for each class, Cls, as follows:

$$\begin{aligned} \text{CA}^2\text{-HOTA}_\alpha\{\text{Cls}\} &= \sqrt{\frac{\sum_{c \in \{\text{TP}_{\text{Cls}}\}} (\mathcal{A}(c) \cdot \mathcal{C}(c, \text{Cls}))}{|\text{TP}_{\text{Cls}}| + |\text{FN}_{\text{Cls}}| + \sum_{f \in \{\text{FP}\}} \mathcal{C}(f, \text{Cls})}} \end{aligned} \quad (4.40)$$

where TP_{Cls} and FN_{Cls} are those which have a ground-truth class Cls, and the notation $\mathcal{C}(c, \text{Cls})$ is the classification score which prediction c has assigned to the ground-truth class Cls. The final score is calculated by averaging over all classes before averaging over α thresholds.

For datasets with many classes (Yang et al., 2019; Dave et al., 2020) we recommend the use of CA²-HOTA as it adjusts for class bias during evaluation.

Federated HOTA.. TAO (Dave et al., 2020) uses a federated evaluation strategy. Not all objects are annotated in all images. Instead, each image is labeled with the set of classes for which there is confirmed no unannotated objects (for which FPs can be evaluated). Extra predictions of other classes should be ignored as they could be present but unannotated.

We propose a version of HOTA which adapts Eq. 4.40 to federated evaluation (Fed HOTA).

$$\begin{aligned} & \text{Fed-HOTA}_\alpha\{\text{Cls}\} \\ &= \sqrt{\frac{\sum_{c \in \{\text{TP}_{\text{Cls}}\}} (\mathcal{A}(c) \cdot \mathcal{C}(c, \text{Cls}))}{|\text{TP}_{\text{Cls}}| + |\text{FN}_{\text{Cls}}| + \sum_{f \in \{\text{FP}\}} \mathcal{I}(f, \text{Cls}) \cdot \mathcal{C}(f, \text{Cls})}} \end{aligned} \quad (4.41)$$

where $\mathcal{I}(f, \text{Cls})$ is 1 if f is from an image where the class Cls should be counted as false positive, and 0 otherwise.

Confidence-Ranked HOTA.. HOTA operates on final tracking predictions rather than confidence-ranked potential tracks. However, for certain tracking scenarios such as monocular 3D tracking where it is difficult for trackers to accurately localise detections a confidence-ranked version (CR-HOTA) is more suitable. This follows other metrics such as Track mAP (Russakovsky et al., 2015) or sAMOTA (Weng et al., 2020).

When using CR-HOTA trackers must output a confidence score for each detection, k . Detections over the whole benchmark are ordered by decreasing confidence. Looping over the ordered detections, the detection recall score is calculated for each one considering all detections with a higher confidence. For 19 fixed recall values (0.05 to 0.95 in 0.05 intervals), the HOTA score is calculated using Eq. 4.13 and Eq. 4.16, by taking into account all detections with a confidence score higher than the maximum confidence score needed to obtain a recall at that threshold. The final CR-HOTA score is given by:

$$\text{CR-HOTA} = \int_0^1 \frac{\text{HOTA}_k}{\text{DetRe}_k} dk \approx \frac{1}{19} \sum_{k \in \{\frac{0.05}{0.9}, \frac{0.1}{0.95}, \dots\}} \frac{\text{HOTA}_k}{\text{DetRe}_k} \quad (4.42)$$

By integrating the value of HOTA/DetRe over a range of DetRe scores, we obtain a formulation which reduces to the original HOTA score when only evaluating detections above a given threshold. In this case the HOTA score would be the same for all values recall value from 0 to DetRe and would be zero afterward. Note that this formulation is the same as how MOTA is adapted to sAMOTA in (Weng et al., 2020).

4.9 Analysing Previous Evaluation Metrics.

In this section, we analyse the previous evaluation metrics MOTA (Bernardin & Stiefelhagen, 2008), IDF1 (Ristani et al., 2016) and Track-mAP (Russakovsky et al., 2015), identifying a number of drawbacks for each one and drawing comparisons to our HOTA metric. See Sec. 4.4 for descriptions of each of the previous metrics. Note that our analysis of each metric is with respect to the properties of detection, association

	MOTA	IDF1	Track-mAP	HOTA
Representation	Final Tracks	Final Tracks	Potential Tracks with Conf. Score	Final Tracks
Matching Mechanism	Bijjective	Bijjective	Highest Conf.	Bijjective
Matching Domain	Detection	Trajectory	Trajectory	Detection
Association Domain	Prev. One Det	Matched Dets	Matched Dets	All Dets
Scoring Function	$1 - \frac{\sum \text{Err}}{ \text{GTDet} }$	F1 Score	Av. Precision	Doub. Jaccard
Bias Toward	Detection	Association	Association	Balanced

Table 4.1: An overview of different design choices and properties for each of the previously used three metrics and HOTA.

and localization. This is not the only framework in which results can be analysed, but one that is common within the tracking community (Leichter & Krupka, 2013).

High-level Comparison Between Metrics. Table 4.1 shows an overview of the key differences between the four metrics.

HOTA can be thought of as a middle ground between MOTA and IDF1. MOTA performs both matching and association scoring at a local detection level which biases scores toward measuring detection, while IDF1 performs both at a trajectory level which ends up being biased towards association. HOTA matches at the detection level while scoring association globally over trajectories. This results in HOTA being balanced between measuring detection and association, exhibiting many of the benefits of each method without the drawbacks.

Track-mAP is similar to IDF1 in many ways in that it performs both matching and association at a trajectory level and as such is biased toward measuring association. However, Track-mAP differentiates itself in that it operates on potential tracks with confidence scores rather than final tracks and does not perform bijective mapping but matches based on the highest confidence valid matches.

4.9.1 Problems with MOTA

MOTA has been the main MOT evaluation metric since 2006. It has served the community over the years, however we believe its drawbacks have restricted tracking research. Now that we are equipped with better tools and a better understanding of the tracking task, we are able to analyse all of the problems with MOTA, and ensure that new metrics, such as HOTA, do not have the same issues. We hoped HOTA will quickly replace MOTA as the default standard for evaluating MOT algorithms, and indeed this has happened. HOTA has since become the default MOT evaluation metric for almost all benchmarks including KITTI and MOTChallenge.

Below we highlight 9 separate problems of the MOTA metric, and show how these problems are addressed in HOTA.

Problem with MOTA 1 *Detection performance significantly outweighs association performance.*

MOTA measures detection errors as FNs and FPs, and association errors as IDSWs. The ratio of the effect of detection errors to association errors on the final score is given by $|FN| + |FP| : |IDSW|$.

For real trackers this ratio is extremely high. For the top ten trackers on the MOT17 benchmark (Milan et al., 2016a) on the 1st April 2020 this ratio varies between 42.3 and 186.4, with an average of 98.6. This is not because trackers are 100 times better at detection than association, but rather that MOTA is heavily biased towards measuring detection. In fact, on average the effect of detection on the final score is 100 times as large as the effect of association.

We also compared the MOTA and MODA (MOTA without IDSWs, which only measures detection) scores for 175 trackers on the MOT17 benchmark. When fitting a linear regression model between MOTA and MODA the R^2 value is 99.4 indicating that the detection only score MODA explains more than 99% of the variation in final MOTA score. In contrast, the R^2 value between MOTA and the number of IDSWs is only 23.7.

This can potentially have significant negative effects on the tracking community. If researchers are tuning their trackers to optimise MOTA to increase scores on benchmarks, then such trackers will be tuned toward performing well for detection while mostly ignoring the requirement of performing successful association.

HOTA is designed to not have this problem as it is composed of a detection and association score which both contribute equally. Tab. 4.2 and Fig. 4.13 show that the numerical values for each of these scores are similar to one another for real trackers.

Problem with MOTA 2 *Detection Precision significantly outweighs the effect of the Detection Recall.*

Let us consider MOTA without considering IDSWs. This is MODA (multi-object detection accuracy). The equation for MOTA/MODA as shown in Eq. 4.2 can be rearranged as follows:

$$\begin{aligned} \text{MODA} &= 1 - \frac{|FN| + |FP|}{|gtDet|} \\ &= \frac{|TP| - |FP|}{|TP| + |FN|} \\ &= \text{DetRe} \cdot \left(2 - \frac{1}{\text{DetPr}}\right) \end{aligned} \tag{4.43}$$

It can be seen that this is not symmetric in terms of recall and precision. The score increases linearly with increasing recall and hyperbolically with increasing precision.

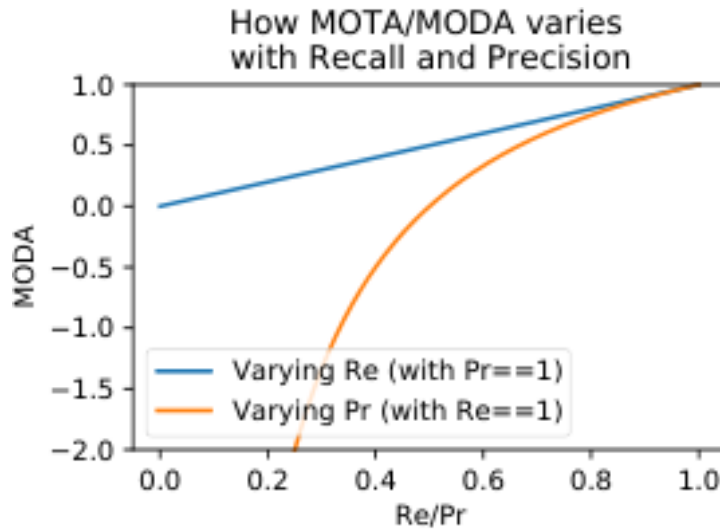


Figure 4.6: Figure showing how MOTA and MODA vary with the Recall and Precision.

This relation can be seen visually in Fig. 4.6. Poor precision has a much greater effect on the final score than poor recall.

Again this promotes researchers to tune trackers to optimise precision at the cost of recall, because poor precision values are penalised extremely heavily by MOTA. This can be seen starkly in Fig. 4.13 where precision values for all trackers are much higher than recall values.

As described in Sec. 4.6, for different tracking applications the importance of recall vs precision varies. For applications such as surveillance, recall is often much more important than precision, and as such, the bias imparted by MOTA is particularly harmful. Ideally a benchmark designed for evaluating trackers for a range of applications would evenly weight precision and recall. HOTA solves this issue by using a symmetric Jaccard formulation and thus ensuring that precision and recall are weighted evenly.

Problem with MOTA 3 *Association errors in MOTA, measured in the form of IDSWs, only take into account short-term (or first order) association.*

In MOTA, an IDSW is a TP which has a prID that is different from the prID of the previous TP (that has the same gtID). Formally a TP, c , is an IDSW under the following criteria:

$$\begin{aligned}
 c \in \text{IDSW} \quad & \text{if} \quad \text{prev}(c) \neq \emptyset \\
 & \text{and} \quad \text{prID}(c) \neq \text{prID}(\text{prev}(c)) \\
 \text{prev}(c) = & \begin{cases} \arg \max_k (t(k)), & \text{if } k \neq \emptyset \\ \emptyset, & \text{otherwise} \end{cases} \\
 & k \in \{\text{TP} \mid t(k) < t(c) \wedge \text{gtID}(k) = \text{gtID}(c)\}
 \end{aligned} \tag{4.44}$$

An IDSW only measures if there is an association tracking error to the previous gtDet. This is equal to a single FNA for each TP comparing to the previous TP from the same gtTraj. This can be thought of as a *first-order approximation* to the global association score over the whole trajectory.

This is only able to capture algorithms' ability to perform "short-term tracking", and is unable to evaluate global long-term tracking over a whole video. HOTA in contrast is able to evaluate *higher-order* global tracking by measuring FNAs and FPAs compared to all other detections in matched trajectories.

Problem with MOTA 4 *MOTA does not take into account association precision (ID transfers).*

The transpose of an IDSW is called an ID transfer (IDTR). An IDTR is a TP which has a gtID that is different from the gtID of the previous TP that has the same prID. Whereas IDSWs compare to the previous gtDet, IDTRs compare to the previous prDet. This is a first-order FPA, whereas an IDSW is a first-order FNA. ID transfers commonly occur when a predicted track spreads over two ground truth tracks. MOTA does not at all penalise such errors.

As an example consider a short video with only two frames. In scenario A, there is one ground-truth object present in both frames, the tracker correctly detects it in each frame but splits it into two tracks. In this case an IDSW occurs and the MOTA score is 0.5. In scenario B, there are two ground-truth objects which are present only in one of the frames each, the tracker again detects both correctly but predicts that they are the same object. In scenario B an ID transfer has occurred, but this is not an IDSW and is not penalised and the MOTA score is a perfect 1.0.

Earlier we saw how MOTA was not symmetric between detection precision and recall. Now we see that it also is not symmetric between association precision and recall. In fact it does not measure association precision (ID transfer) errors at all.

This is again potentially extremely undesirable behaviour. Trackers can take advantage of this fact to "hack the metric" to improve their score while performing worse tracking by artificially merging their trajectories over multiple ground-truth objects.

HOTA solves this error by measuring both FPAs and FNAs when measuring association accuracy.

Problem with MOTA 5 *MOTA does not reward trackers that correct their own association mistakes. In fact, it unfairly penalises such corrections.*

MOTA is not able to successfully evaluate tracking when a tracker corrects itself after making an association mistake. In this case, MOTA will penalise the tracker twice, first for making a mistake, and then for correcting it. An example of this can be seen in Fig. 4.7. In this example the ordering of the scores should be (A)>(B)>(C). This is because (A) corrects itself when making a tracking mistake. (B) does not correct itself

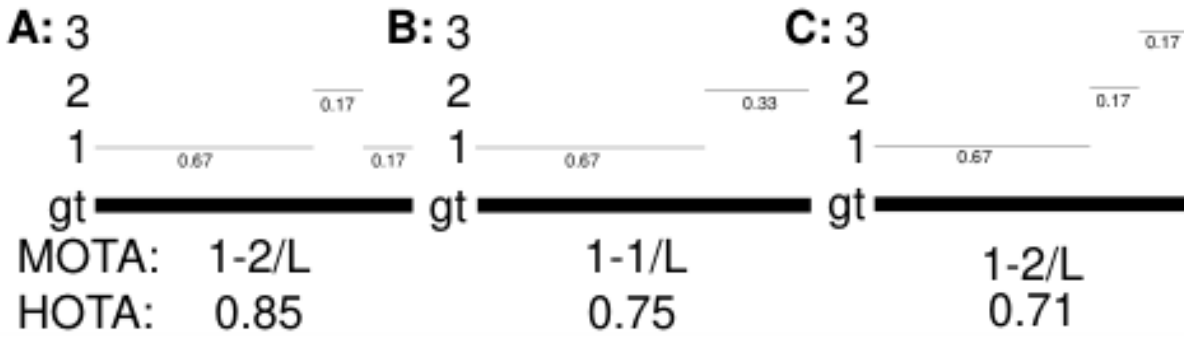


Figure 4.7: An example showing how MOTA does not correctly evaluate tracking situations when a tracker corrects itself after making a mistake. The thick line is the gtTraj. Thin lines are prTrajs. All detections are TPs.

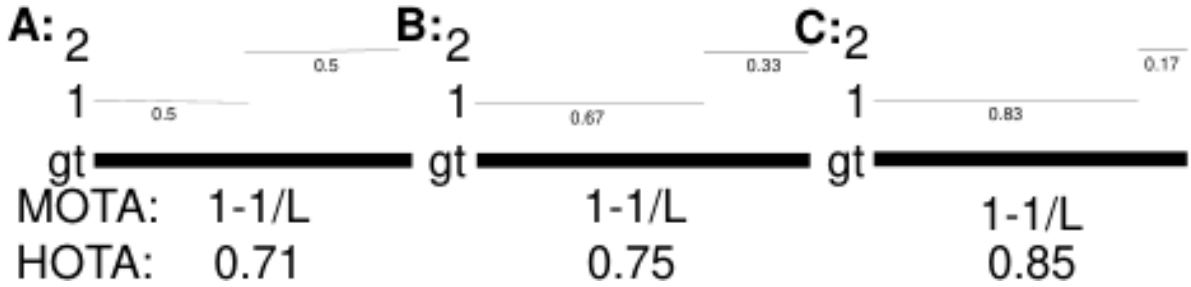


Figure 4.8: An example showing how MOTA does not reward trackers for having a greater alignment between predicted and ground-truth trajectories. The thick line is the gtTraj. Thin lines are prTrajs. All detections are TPs.

and continues to track the object with the wrong ID. (C) is even worse making a further tracking mistake. However, the MOTA score does not follow this intuitive ranking. MOTA is unable to account for the tracker correcting itself and counts the correction as a further mistake. This means that under MOTA (A) and (C) are equal even though (A) is clearly much better than (C). This property of MOTA heavily disincentivises research into long-term trackers that are able to correct from mistakes.

HOTA solves this issue by measuring the association globally over the whole sequence. In the example HOTA ranks the trackers with the intuitive ranking of (A)>(B)>(C).

Problem with MOTA 6 *MOTA does not reward trackers for having a greater alignment between predicted and ground-truth trajectories.*

Another problem that arises because MOTA only evaluates first-order short-term tracking, is that it does not reward trackers based on how well predicted trajectories and ground-truth trajectories align. This can be seen clearly in Figure 4.8, where it is clear that (C)>(B)>(A) due to the fact that in (C) one of the prTrajs is 83% similar

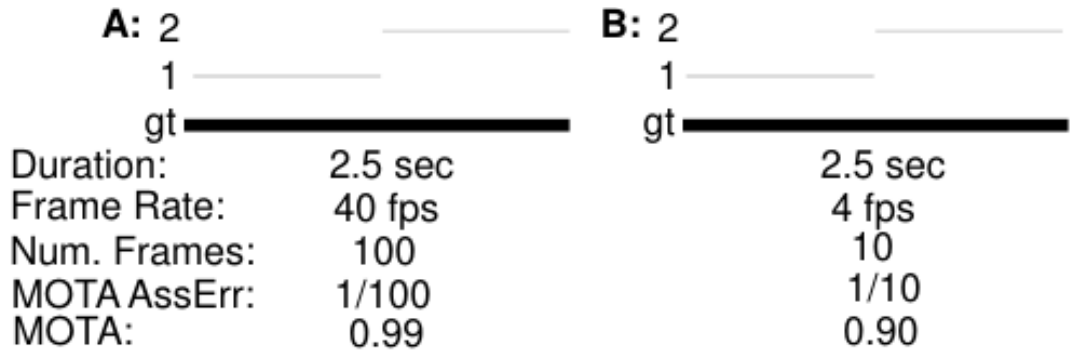


Figure 4.9: An example on MOTAs variability with the frame rate. The thick line is the GT trajectory. Thin lines are predicted trajectories. All detections are TPs.

to the gtTraj, whereas in (A) at best one of the trajectories is only 50% similar to the gtTraj. However, because MOTA only evaluates short-term association, it is unable to differentiate between these cases and correctly measure the trajectory alignment.

This property is important, because as prTrajs and gtTrajs become more similar to one another the evaluation score should increase.

HOTA solves this problem by measuring association globally across the whole sequence. Thus HOTA correctly ranks these trackers, and is able to take into account the level of alignment between trajectories.

Problem with MOTA 7 *In MOTA, the influence of association (IDSWs) on the final score is highly dependent on the video frame rate.*

This is easily seen with an example, as shown in Figure 4.9, where we show exactly the same sequence processed by the same tracker. In (A), however, it is evaluated at the original video frame rate of 40 fps, in (B) it is evaluated at a reduced frame rate of 4 fps. Different scenarios call for different frame rates, e.g., in a surveillance scenario the frame rate of the cameras tends to be low. In Figure 4.9 we show how the tracker is performing identically in both scenarios. However, as can be seen, the MOTA score reflects a poorer performance of the tracker when used at a low frame rate, 0.90 MOTA vs 0.99. This occurs because MOTA registers only a single error in both cases (one IDSW), and yet this is normalised by the total number of GT detections over the video, which increases with the number of frames.

Our HOTA does not have this problem, because it measures the association error for each TP and averages this over the TPs. Thus HOTA is independent of the frame rate, and the HOTA value will be the same (0.707) for both (A) and (B) in Figure 4.9.

Problem with MOTA 8 *MOTA does not take into account localisation accuracy.*

MOTA is calculated at a pre-set value of α for determining the matching between prDets and gtDets, but the value of MOTA is the same regardless of how correct these

matches are as long as they are over a minimum localisation threshold. Thus, MOTA was proposed as one of two metrics that should be used in combination for measuring MOT accuracy. The second of these is MOTP (Multi-Object Tracking Precision), which is simply the average localisation similarity over the TP matches (ignoring detection and association errors). Since MOTA was designed to be used together with MOTP, it is not able to measure localisation together in a single metric with detection and association.

HOTA solves this issue by including the localisation accuracy into its calculation. By calculating the score over a range of α values, HOTA is able to include localisation along with detection and association together into one score.

Problem with MOTA 9 *MOTA scores can be negative and are unbounded.*

The final, and perhaps most frustrating, problem of MOTA is that scores are not between 0 and 1, as is typically expected for evaluation metrics. Although the maximum MOTA score is 1 for perfect tracking, there is no lower limit to the MOTA score, and it can go down to negative infinite. This is caused by the score decreasing linearly with the number of FPs, which can be continuously added forever. This leads to a score that is hard to interpret: how can we understand a negative MOTA score? Furthermore, a MOTA close to zero might not be as bad as it seems. HOTA does not have this problem by conveniently being a score between 0 and 1.

4.9.2 Problems with IDF1

The IDF1 metric ([Ristani et al., 2016](#)) was originally designed for evaluating tracking in a multi-camera setting, but is trivial to apply to a standard single camera setting. An overview of how IDF1 is calculated can be found in Sec. 4.4.

IDF1 was designed to measure the concept of “identification”. This concept is related to, but distinct from the concepts of detection and association which we analyse in this chapter. Identification is more about determining which trajectories are present, rather than detecting objects and associating them throughout time.

In recent years IDF1 has been adopted by a number of MOT papers ([Maksai et al., 2017](#); [Maksai & Fua, 2019](#); [Wang et al., 2019a](#)) for MOT evaluation instead of MOTA, as these papers wish to adequately account for association in evaluation, which is lacking from MOTA. However, IDF1 produces counter-intuitive and non-monotonic results for measuring detection. Due to this, no single-camera MOT benchmarks have adopted IDF1 as the main metric for evaluating trackers and new benchmarks are still choosing to use MOTA instead of IDF1 despite all of MOTA's drawbacks ([Voigtlaender et al., 2019b](#); [Sun et al., 2020](#); [Wang et al., 2020](#); [Yu et al., 2020](#)). This is because detection is such an important part of tracking evaluation, and IDF1 is not able to adequately measure it.

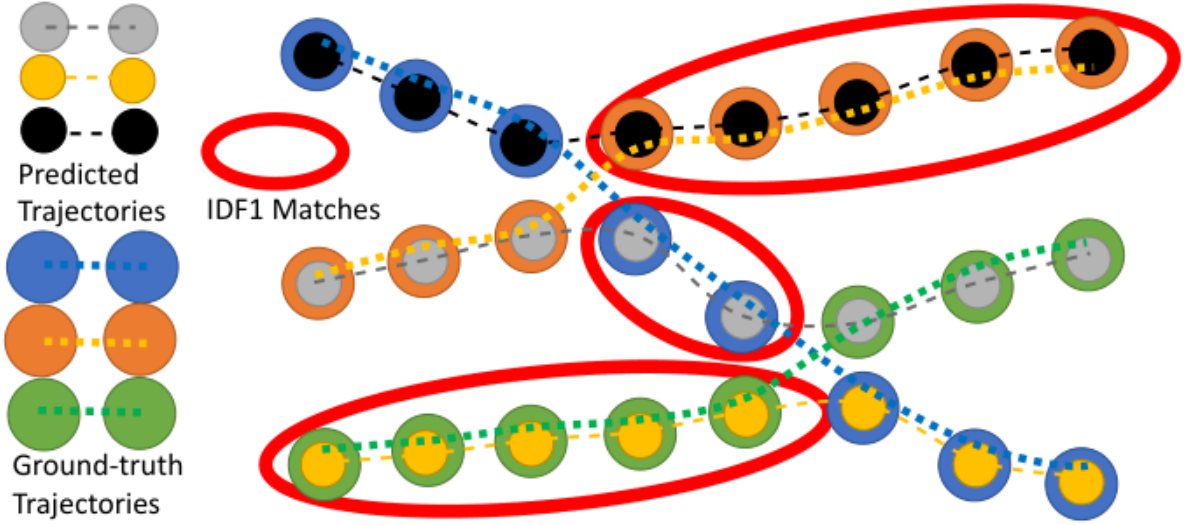


Figure 4.10: A tracking example which shows how the single best trajectory matching, as performed by IDF1, can result in unintuitive matches between trajectories.

In IDF1, each gtTraj can be matched with a single prTraj and vice versa. This contrasts to HOTA where gtTrajs and prTrajs are evaluated as being matched if they are matched at any point in time at any detections.

Just like MOTA can be thought of as only measuring *first-order association* (a single previous association for each detection) compared to HOTA measuring *higher-order association* (all possible associations for each detection), IDF1 can be thought of as measuring *first-order matching* (a single possible match for each trajectory), compared to HOTA measuring *higher-order matching* (all possible matches for each trajectory).

Because IDF1 only allows a single best set of matching trajectories to be evaluated, any trajectory that does not end up in this matching set is counted as a negative and decreases the score, even if it contributes correct detections and associations.

Below we highlight 5 separate problems of the IDF1 metric, and show how these problems are addressed in HOTA.

Problem with IDF1 1 *The best unique bijective mapping of whole trajectories is often not representative of the actual alignment between ground-truth and predicted trajectories.*

This can be best understood with an example, as shown in Fig 4.10. In this example the best bijective mapping matches the blue gtTraj with the grey prTraj. This occurs even though this gtTraj is better matched with both other prTraj, and this prTraj is better matched with both other gtTraj. Thus in this case, matching these trajectories is the worst possible matching for both the gtTraj and the prTraj. However they are still matched here, as when considering all trajectories, all of the better matching options are better matched elsewhere and thus this worst fitting match is what ends up being evaluated. This is a perfect example to show why a unique bijective mapping of

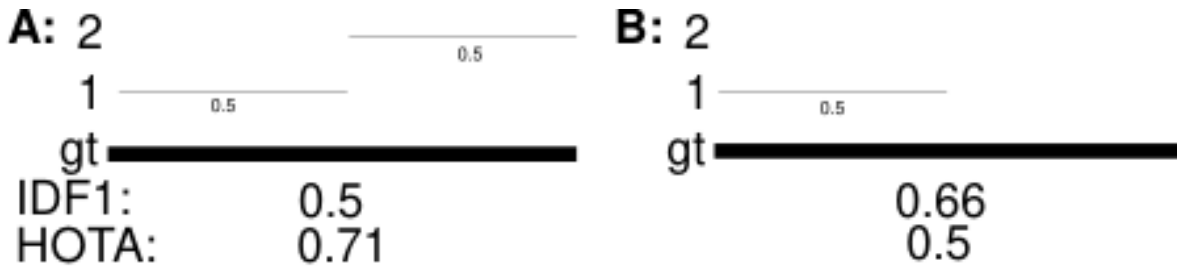


Figure 4.11: A simple tracking example showing how IDF1 fails to correctly rank tracking performance because the score can decrease with improving detection. The thick line is the gtTraj. Thin lines are prTrajs. All detections are TPs, except the second half of the gt in B which are FNs.

trajectories does not make intuitive sense for evaluating tracking, which often results in complex overlaps between ground-truth and predicted trajectories.

HOTA avoids this problem by not forcing a single global matching between trajectories but rather evaluating over all combinations of ground-truth and predicted trajectories that overlap at any point.

Problem with IDF1 2 *IDF1 actually decreases with improving detection.*

In Fig. 4.10 it can be seen that in non-matched regions there are many correct detection results. These correct detections do not add positively to the final score. In fact, each one of these correct detections decreases the final IDF1 score. Thus IDF1 is non-monotonic in detection.

This can be seen more clearly in Fig. 4.11, where the score for (A) should be higher than for (B), but IDF1 ranks the two tracking results in the other order because it penalises the correct detections in the second trajectory.

This property is extremely counter productive for many tracking scenarios such as in autonomous driving where it is critical to correctly detect objects.

HOTA does not have this problem and is strictly monotonic in detection such that improving detection always improves the HOTA score.

Problem with IDF1 3 *IDF1 ignores the effect of how good association is outside of matched sections.*

Due to the fact that only the matched sections count towards the score, the association can be trivially bad in non matched regions and this will have no effect on the IDF1 score. Thus creating better or worse association does not necessarily correlate to an increase or decrease in the IDF1 score. This can be seen most clearly in Fig. 4.12 where the IDF1 scores for (A), (B) and (C) are identical where it is clear that the trackers should be evaluated such that (A) > (B) > (C).

HOTA does not have the same problem as all detections are evaluated, not just the best matching ones. Thus HOTA ranks these trackers correctly.

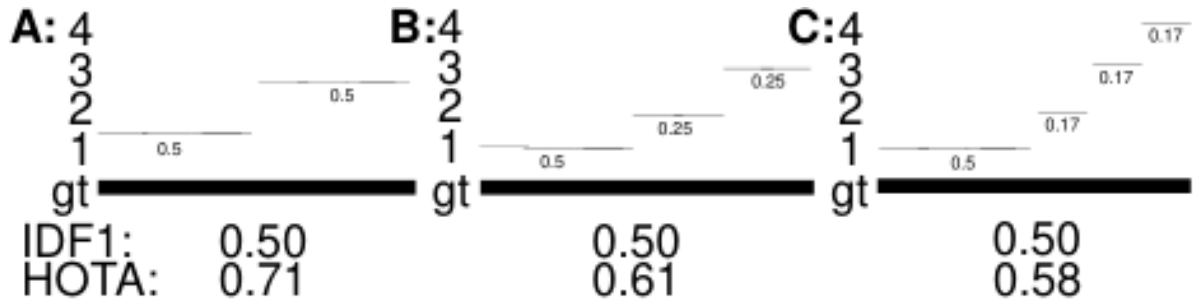


Figure 4.12: A simple tracking example showing how IDF1 fails to correctly rank tracking performance because it ignores the effect of any association that is not included in the best matching trajectories. The thick line is the gtTraj. Thin lines are prTrajs. All detections are TPs.

Problem with IDF1 4 *Scoring highly on IDF1 is more about estimating the total number of unique objects in a scene than it is about good detection or association.*

Due to the fact that any extra trajectories that are not matched as one of the best matching trajectories are automatically counted as negatives, one of the key design goals for trackers that are optimizing for IDF1 becomes to estimate the total number of unique objects in the scene and only produce that many tracks, rather than performing good detection or good association. This is because if there are more (or less) predicted trajectories than ground-truth trajectories, the extra (missing) trajectories are automatically counted as negatives and severely decrease the score. This is a very different objective than the objective defined in this chapter for multi-object tracking, which is to detect the presence of objects and to associate these consistently over time.

In contrast, optimizing for HOTA directly optimises for both accurate detection and accurate association as the final HOTA score is a combination of scores for each of these components.

Problem with IDF1 5 *IDF1 does not evaluate the localisation accuracy of trackers.*

Like MOTA, IDF1 is also evaluated at a fixed α threshold for how accurate localisation needs to be for detections to match, however the actual localisation of the detections is ignored as long as they are beyond the threshold.

HOTA is evaluated over a range of localisation thresholds α and as such HOTA increases as the localisation of trackers increases.

4.9.3 Problems with Track-mAP

Track-mAP is an extension of the mAP metric, commonly used for evaluating detection (Everingham et al., 2010; Lin et al., 2014; Russakovsky et al., 2015), to the video domain. An overview of Track-mAP can be found in Sec. 4.4. Track-mAP does not operate on

final tracking results but on confidence-ranked potential tracking results. This results in it being non-trivial to compare to other metrics as it operates on a different, metric specific tracking output format. Track-mAP is similar to IDF1 in that it also performs matching at a trajectory level. This results in it also being non-monotonic in detection. Below we highlight 5 problems of Track-mAP, and show how these are addressed in HOTA.

Problem with Track-mAP 1 *The interpretation of tracking outputs is not trivial, nor easily visualisable.*

With other metrics, when one wishes to understand a tracker, all they have to do is visualise the tracking results. One can easily identify each of the error types defined for each of the other metrics. This is not the case for Track-mAP. Here the output is likely many overlapping outputs, many of which have low confidence scores, with the actual influence of each trajectory on the final score being hidden behind the implicit confidence ranking. This makes developing trackers that optimise Track-mAP a potentially frustrating experience. It also makes user-comparison, like the type we perform in Sec. 4.11 impossible, as the representation does not allow meaningful analysis of the visual results.

Problem with Track-mAP 2 *It is possible to game the metric by tuning trackers for quirks of the metric, which do not necessarily correspond to better tracking.*

There are a number of ways in which a trackers output can be tuned to increase Track-mAP without actually improving the tracking result. One of these, as discussed in (Luiten et al., 2019), is that by producing many different predictions with low confidence scores, it greatly increases the chances of obtaining a correct trajectory match and thus improving the score. In (Luiten et al., 2019) they do this by replicating trajectories for each class for all other classes. Since this is possible to tune algorithms in this way, it becomes a requirement for methods to adequately compete. Trackers that do not do this will be heavily penalised. Some benchmarks (Dave et al., 2020) have attempted to mitigate this issue by restricting results to a maximum number of trajectories per video. However (Dave et al., 2020) still allows 300 trajectories per frame which still enables significant gaming. HOTA and other metrics do not have this issue because trackers are required to produce final tracking results.

Problem with Track-mAP 3 *The threshold for being counted as a positive match is so high that a lot of improvement in detection, association and localisation is ignored by the metric.*

In the Track-mAP version used in (Russakovsky et al., 2015; Zhu et al., 2020) for a trajectory to be counted as a positive it must be successfully detected and associated such that the Jaccard over detections is at least 50%. This is a very high threshold for many tracking scenarios. We can see in Fig. 4.13 that for all published trackers on

the MOTChallenge leader-board the average Jaccard association alignment between trajectories (AssA) ranges between 0.25 and 0.5. This means that even for the very best tracker, more than half of its best guess predictions will be counted as errors in Track-mAP. This has the effect that as trackers improve significantly in terms of detection and association this is not shown by an improvement in metric scores. E.g. trajectories that align 5% or 45% are given the same score. For the Track-mAP version used in (Yang et al., 2019; Dave et al., 2020) this threshold is even harder to reach because although the threshold is still 50%, the score that must be above this threshold is effectively a multiplication of both the trajectory alignment and the average localisation score across the trajectory which is usually much lower (see Sec. 4.4)

HOTA does not have this issue as it measures the alignment between all trajectory pairs, not just those over a certain threshold.

Problem with Track-mAP 4 *Improving detection (adding correctly matching trajectories) can decrease the score.*

Track-mAP is non-monotonic in detection. This is because it matches at a trajectory level (like IDF1) such that extra trajectories are counted as negatives, even if they contain correct, previously unaccounted for detections. As described previously this is a non-intuitive and undesirable property for many applications such as autonomous driving where detection is critical. HOTA on the other hand is monotonic in detection.

Problem with Track-mAP 5 *Track-mAP mixes association and detection (and localisation) in a way that is not error type differentiable.*

In the Track-mAP version used in (Russakovsky et al., 2015; Zhu et al., 2020) the score used to measure whether trajectories match is a combination of both detection scores and association scores in a way that is not separable or interpretable. Thus trajectories can match and add positively to the score if they have high detection accuracy and medium association accuracy, or medium detection accuracy and high association accuracy. The Track-mAP metric does not give any indication as to which of these situations is occurring and as such has very limited use for understanding and optimizing the behaviour of trackers. The Track-mAP version used in (Yang et al., 2019; Dave et al., 2020) is even worse in that the matching score also mixes localisation accuracy with detection and association. Furthermore the effect of localisation accuracy of the score can vary based on differences in object scale over time, which is even more unintuitive. Note that the detection version of mAP (Everingham et al., 2010; Lin et al., 2014; Russakovsky et al., 2015) which the tracking version is based upon does not have this problem because the matching score only measures a single type of error, which is detection errors. The problem arises when this metric is extended to video to measure multiple error types simultaneously.

HOTA does not have this problem because it is designed to be decomposable into separate scores for each error type, such that the effect of each error type on the final metric is clear and the overall metric is error type differentiable.

4.9 Analysing Previous Evaluation Metrics.

	Ranking			Scores			Sub-Scores					
	HOTA	MOTA	IDF1	HOTA	MOTA	IDF1	DetA	AssA	DetRe	DetPr	AssRe	AssPr
MPNTrack17 (Brasó & Leal-Taixé, 2020)	1	2 (↑ 1)	1 (-)	46.6	55.7	59.1	46.2	47.3	49.2	75.9	52.8	70.2
eTC17 (Wang et al., 2019a)	2	6 (↑ 4)	2 (-)	45.1	51.9	58.1	44.1	46.4	47.5	72.8	51.1	71.6
Tracktor++v2 (Bergmann et al., 2019)	3	1 (↓ 2)	6 (↑ 3)	45.1	56.3	55.1	45.3	45	47.3	79.1	48.1	78.2
eHAF17 (Sheng et al., 2018)	4	7 (↑ 3)	7 (↑ 3)	43.6	51.8	54.7	43.6	44	46.9	73.4	49.3	69.8
SAS_MOT17 (Maksai & Fua, 2019)	5	31 (↑ 26)	3 (↓ 2)	43	44.2	57.2	37.5	49.6	40	72.8	53.2	75.8
YOONKJ17 (Yoon et al., 2020)	6	8 (↑ 2)	10 (↑ 4)	42.9	51.4	54	43	43.1	46	74.1	48	70.4
DMAN (Zhu et al., 2018)	7	21 (↑ 14)	5 (↓ 2)	42.7	48.2	55.7	39.9	46	42.4	73.2	49.6	72.7
jCC (Keuper et al., 2018)	8	11 (↑ 3)	8 (-)	42.6	51.2	54.5	41.6	44.1	44.3	73.1	46.4	77.6
NOTA (Chen et al., 2019)	9	10 (↑ 1)	9 (-)	42.6	42.6	54.5	41.9	43.5	44.2	75.1	47.4	72.7
Tracktor++ (Bergmann et al., 2019)	10	3 (↓ 7)	12 (↑ 2)	42.5	53.5	52.3	43.4	42	45.3	77.8	45.4	75.7
STRN_MOT17 (Xu et al., 2019)	11	12 (↑ 1)	4 (↓ 7)	42.4	50.9	56	41.6	43.5	44.2	73.3	47	72.3
JBNOT (Henschel et al., 2019)	12	4 (↓ 8)	13 (↑ 3)	41.5	52.6	50.8	44	39.5	47.3	73.5	42.7	73.6
MOTDT17 (Long et al., 2018)	13	13 (-)	11 (↓ 2)	41.5	50.9	52.7	41.9	41.3	44.5	74.4	45.3	70.2
EDMT17 (Chen et al., 2017)	14	15 (↑ 1)	15 (↑ 1)	41.4	50	51.3	42.3	40.8	45.4	73.4	43.1	77.5
MHT_bLSTM (Kim et al., 2018)	15	24 (↑ 9)	14 (↓ 1)	41.1	47.5	51.9	40	42.5	42.5	74.4	46.9	73.5
AM_ADM17 (Lee et al., 2018)	16	22 (↑ 6)	13 (↓ 3)	40.7	48.1	52.1	39.9	41.8	42.4	73.9	44.9	73.8
HAM_SADF17 (Yoon et al., 2018)	17	20 (↑ 3)	16 (↓ 1)	40.5	48.3	51.1	39.8	41.4	42.1	75	44.1	75.7
PHD_GSDL17 (Fu et al., 2018)	18	23 (↑ 5)	18 (-)	39.4	48	49.6	40.1	39	42.5	74.6	43.1	70.2
FWT_17 (Henschel et al., 2018)	19	9 (↓ 10)	22 (↑ 3)	39.2	51.3	47.6	42.6	36.4	45.2	74.9	39.4	71.9
FAMNet (Chu & Ling, 2019)	20	5 (↓ 15)	19 (↓ 1)	39	52	48.7	41.7	36.8	43.7	75.9	40.6	69.2
MHT_DAM_17 (Kim et al., 2015)	21	14 (↓ 7)	23 (↑ 2)	38.9	50.7	47.2	41.9	36.5	44.5	75.1	38	80
OTCD_1_17 (Liu et al., 2019a)	22	19 (↓ 3)	21 (↓ 1)	38.6	48.6	47.9	40	37.5	42.1	75.5	40.2	73.8
FPSN (Lee & Kim, 2018)	23	28 (↑ 5)	20 (↓ 3)	38.2	44.9	48.4	38.8	38	41.8	71.9	41.3	70.2
GMPHDOGM17 (Song et al., 2019)	24	16 (↓ 8)	24 (-)	38.2	49.9	47.1	41.7	35.2	44.2	74.9	40.8	61.5
MTDF17 (Fu et al., 2019)	25	17 (↓ 8)	26 (↑ 1)	37.8	49.6	45.2	42.5	34.2	45.8	71.8	36.1	75.9
MASS (Karunasekera et al., 2019)	26	25 (↓ 1)	25 (↓ 1)	36.9	46.9	46	39	35.2	41.6	73.1	38.4	70.2
LM_NN_17 (Babaee et al., 2019)	27	27 (-)	27 (-)	36.6	45.1	43.2	37.2	36.4	38.7	78.3	37.9	81.2
PHD_GM (Sanchez-Matilla & Cavallaro, 2019)	28	18 (↓ 10)	28 (-)	36.2	48.8	43.2	41.1	32.1	43.7	74.2	34.9	69.5
EAMTT_17 (Sanchez-Matilla et al., 2016)	29	35 (↑ 6)	29 (-)	34.7	42.6	41.8	36.8	33.2	39.2	72.2	35.5	72.7
SORT17 (Bewley et al., 2016)	30	34 (↑ 4)	30 (-)	34.1	43.1	39.8	37.4	31.4	39.8	73.7	32.8	79.4
IOU17 (Bochinski et al., 2017)	31	26 (↓ 5)	31 (-)	33.7	45.5	39.4	38	30.2	40.1	74.8	32.6	73.7
HISP_T17 (Baisa, 2018)	32	29 (↓ 3)	33 (↑ 1)	33.4	44.6	38.8	38.7	29.2	41.1	74	30.7	76.1
GMPHD_SHA (Song & Jeon, 2016)	33	33 (-)	32 (↓ 1)	33.2	43.7	39.2	37	30.2	39.3	73.2	31.8	76.9
GMPHD_DAL (Baisa, 2019)	34	30 (↓ 4)	35 (↑ 1)	31.5	44.4	36.2	38	26.4	40	75.3	28.9	69.6
GM_PHD_D (Eiselein et al., 2012)	35	32 (↓ 3)	36 (↑ 1)	30.7	44	34.2	38	25.1	40	75.3	26.3	75.9
GMPHD_KCF (Kutschbach et al., 2017)	36	37 (↑ 1)	34 (↓ 2)	30.4	39.6	36.6	35.9	26.1	39.4	67.2	27.6	72.4
GMPHD_N1Tr (Baisa & Wallace, 2019)	37	36 (↓ 1)	37 (-)	30.4	42.1	33.9	36.4	25.7	38.2	75.7	27	76.1

Table 4.2: Results on MOT17 for all published (peer reviewed) trackers.

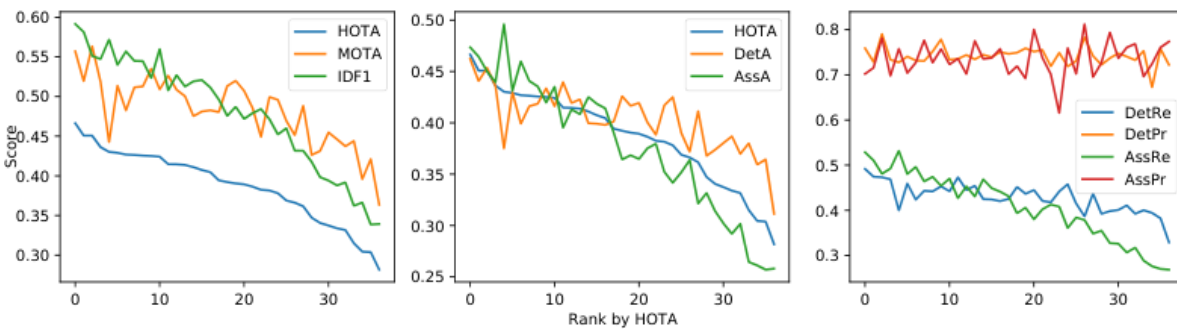


Figure 4.13: Trends between various metrics and sub-metrics on MOT17. All 37 trackers from Table 4.2 are shown in order of decreasing HOTA.

4.10 Evaluating Trackers with HOTA on MOTChallenge

In order to see how HOTA compares to other metrics for real state-of-the-art trackers, we evaluated HOTA on trackers submitted to the *MOTChallenge* MOT17 benchmark, and compared these HOTA scores with the MOTA and IDF1 scores. We cannot compare to Track-mAP because this metric requires trackers to supply confidence scores which is not the case for the MOTChallenge benchmark.

We restrict our evaluation to only those methods that are published in peer reviewed journals and conferences. We evaluate 37 different trackers (Eiselein et al., 2012; Kim et al., 2015; Bewley et al., 2016; Sanchez-Matilla et al., 2016; Song & Jeon, 2016; Bochinski et al., 2017; Chen et al., 2017; Kutschbach et al., 2017; Baisa, 2018; Fu et al., 2018; Henschel et al., 2018; Keuper et al., 2018; Kim et al., 2018; Lee & Kim, 2018; Lee et al., 2018; Long et al., 2018; Sheng et al., 2018; Yoon et al., 2018; Zhu et al., 2018; Babaei et al., 2019; Baisa, 2019; Baisa & Wallace, 2019; Bergmann et al., 2019; Chen et al., 2019; Chu & Ling, 2019; Fu et al., 2019; Henschel et al., 2019; Karunasekera et al., 2019; Liu et al., 2019a; Maksai & Fua, 2019; Sanchez-Matilla & Cavallaro, 2019; Song et al., 2019; Wang et al., 2019a; Xu et al., 2019; Brasó & Leal-Taixé, 2020; Yoon et al., 2020) on MOT17 (Milan et al., 2016a). This includes all of the trackers for which the relevant bibliographic information was available when this analysis was performed on the 1st April 2020.

Ranking Methods by HOTA. In Table 4.2 we show all of the published trackers, ranked according to our proposed HOTA metric. For fine-grained analysis, we also show the detection accuracy DetA, the association accuracy AssA, the detection recall DetRe, the detection precision DetPr, the association recall AssRe and the association precision AssPr. For a definition of these metrics, see Sec. 4.6. We also show the scores for MOTA and IDF1 metrics, and add an indicator to how the rankings change in compared to HOTA ranking.

Comparing Trends across Metrics. In Fig. 4.13 we show how the scores for different metrics and sub-metrics vary across all of the trackers that we evaluated on MOT17.

Fig. 4.13 (left) shows results for HOTA along with the two previously used metrics MOTA and IDF1. We observe, (i) that although these metrics do not always agree, they do follow a similar loose trend. (ii) it is clear according to all three metrics which trackers are performing well and which do not. This is not surprising, as all three metrics are aiming to quantify the quality of tracking. This can also be seen in Tab. 4.2 where the top-performing methods according to each metric is always within the top four methods over all other metrics. (iii) there are some significant differences in ranking between HOTA and both MOTA and IDF1. (iv) in general, HOTA aligns better in ranking with IDF1 than MOTA. This is not surprising, as HOTA and IDF1 both aim to measure long-term tracking quality, whereas MOTA is only able to capture short-term tracking success. This is also reflected in the table, where the change in rankings for MOTA is larger compared to IDF1.

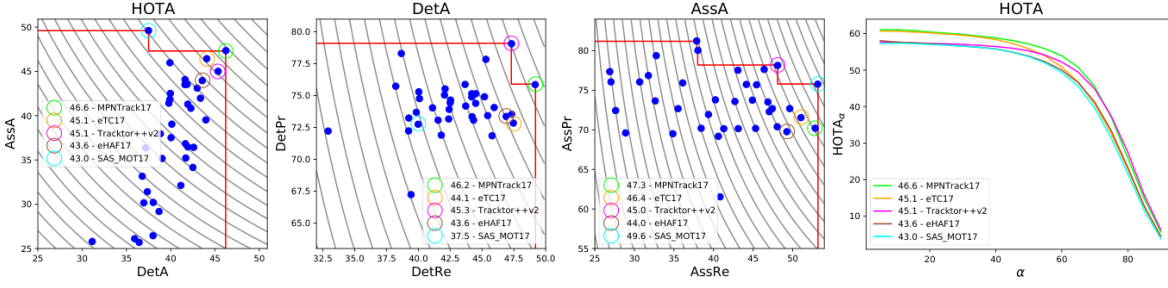


Figure 4.14: Comparison between sub-metrics showing results for all peer reviewed trackers on MOT17. Each of the four plots shows a different decomposition of metrics into their corresponding sub-metrics for evaluating different aspects of tracking. The grey curves are level sets contours of constant score. The red staircase function shows the Pareto front. Only the top-5 trackers by HOTA are shown in the legend and far right plot for clarity.

In Fig. 4.13 (middle), the HOTA score is compared with its two major components, the DetA and AssA scores which measure detection success and association success, respectively. HOTA is computed as the geometric mean of the two and thus is always between the two values. We make the following observations. (i) both detection accuracy and association accuracy improve as trackers get better; (ii) top-performers are better at association than detection, while poor performing trackers are better at detection; (iii) there is larger variability over the association score over different trackers compared to the detection scores. This is expected as all trackers used the given public detections as input proposals.

In Fig. 4.13 (right) we compare four different components of HOTA: the *detection precision* and *recall*, and *association precision* and *recall*. As can be seen, (i) precision values are higher than recall for both detection and association; (ii) precision values are mostly within a similar range across all trackers, whereas recall values show an obvious trend to decrease as the tracker performance is dropping.

Analysing the State-of-the-Art in Multiple Dimensions. HOTA combines the different aspects of tracking in a balanced way suitable for ranking trackers. However it is also informative to compare trackers along all of the different dimensions of tracking. In Fig 4.14 we compare trackers along a number of different dimensions within the sub-metric space of the HOTA family of metrics.

This analysis allows one to clearly see the benefits and pitfalls of certain trackers, and allows for the selection of top performing trackers for different applications that may have different requirements. Any tracker along the multi-dimensional Pareto front can be considered to be state-of-the-art in at least one aspect of tracking performance.

The fourth subplot shows how the HOTA score varies over the localisation threshold α for the top five ranked trackers. By showing performance over the range of all thresholds we are able to analyse and compare different properties of trackers that

are not otherwise apparent by using a single evaluation score, such as which trackers perform very well when matches are allowed to be loosely localized and those that still perform well when a higher standard of localization is required.

What is also clear from this analysis is that the set of lowest-level sub-metrics (DetRe, DePr, AssRe, AssPr) are not enough on their own to tell the whole story about the results between different trackers. One is able to gain a greater level of understanding by examining the higher-level metrics which are combinations of these sub-metrics (DetA, AssA and HOTA). This highlights one of the key benefits of HOTA compared to previous evaluation approaches, that it simultaneously is able to measure different aspects of tracking performance, while being able to combine these together into unified representative scores.

Analyzing Metrics across Detectors. The MOT17 benchmark requires methods to produce results using the same tracking method with a set of three different input detections. Thus it is possible to analyze how different performance metrics behave when using different detectors. Fig. 4.15 shows such an analysis for HOTA, MOTA and IDF1, as well as for all of the sub-metrics of HOTA. For all main metrics, using a better input detector improves the score. Of the three main metrics, MOTA is by the most affected by the choice of input detector. On the other hand, HOTA and IDF1 exhibit similar trends when using different detectors as input. In fact, MOTA exhibits similar trends to DetA. This is because, as discussed in Sec. 4.9.1, MOTA is mostly a proxy for detection accuracy and thus is highly correlated with DetA. As expected, the association scores are far less dependent on the detector input, although it can be seen that better detectors still aid better association. This is not surprising – having more correct detections allows for more correct associations to be made. Precision values for both detection and association are less affected by the choice of detector compared to recall values.

Do the Metrics Disagree where we expect them too? In Sec. 4.9 we laid out a number of theoretical problems of both the MOTA and IDF1 metrics and discussed how HOTA addresses these issues. In that analysis, we argued that MOTA and IDF1 are two ends of the metric spectrum, with HOTA being the middle ground. One of the main issues with MOTA is that it does not adequately score association and mostly only depends on detection accuracy, while IDF1 is exactly the opposite – heavily relying on accurate association while exhibiting non-intuitive behavior with regards to detection quality.

In Fig. 4.16 we plot the MOTA, IDF1 and HOTA scores for all trackers on MOT17 against the DetA and AssA sub-scores, which measure the detection and association accuracy, respectively. We obtain the coefficient of determination (R^2) by fitting the line-of-best-fit and determining the strength of correlation between these metrics. As can be seen, the theoretical analysis of the weaknesses of MOTA and IDF1 is reflected in these results. Our observations are the following. (i) MOTA highly correlates with the detection score (0.96) while exhibiting low correlation with the association

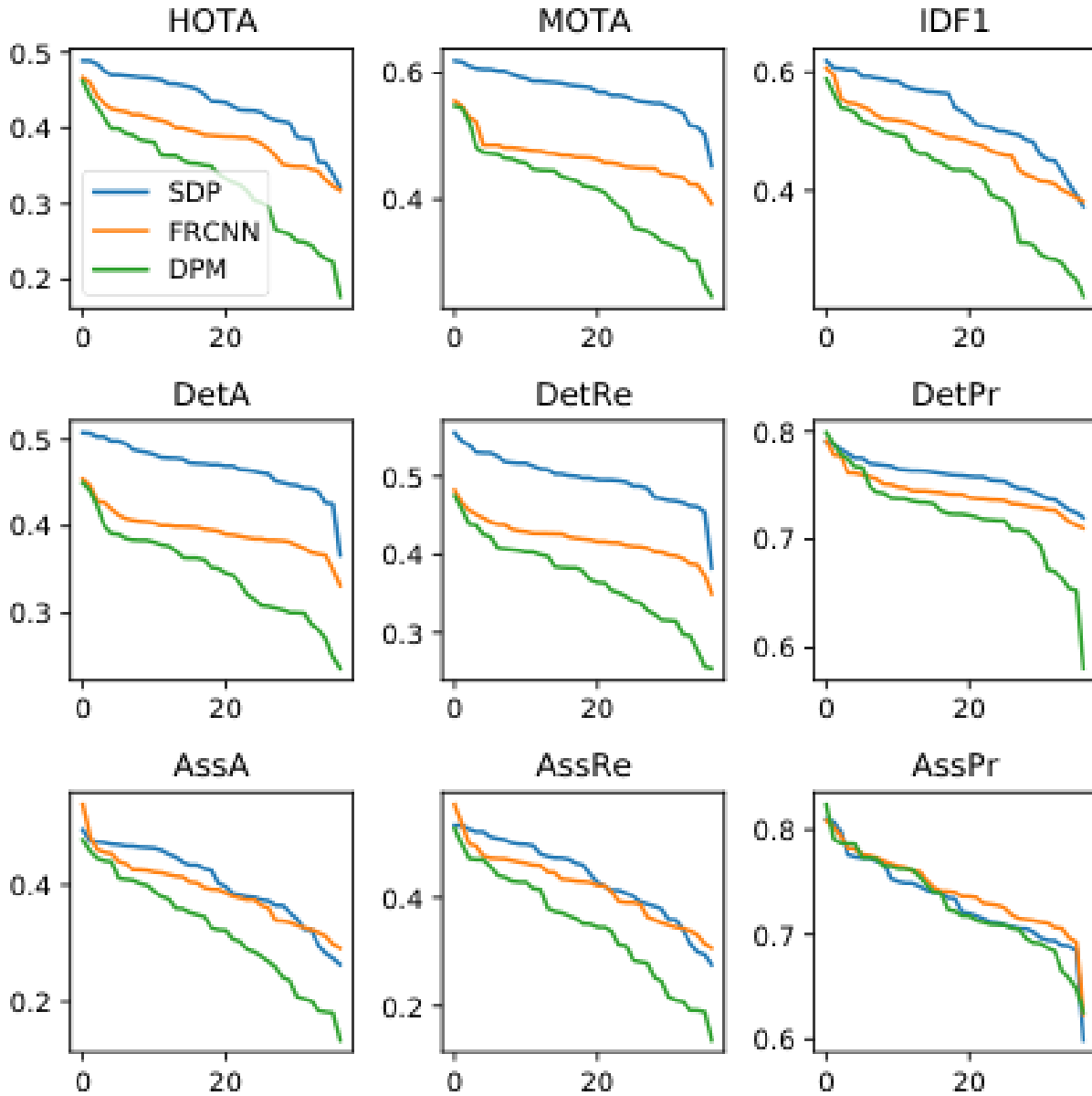


Figure 4.15: Trends for various metrics when using different detectors as input on MOT17. All 37 trackers from Table 4.2 are shown, and are ordered separately for each plot by the metric used.

score (0.46). (ii) IDF1 exhibits almost the opposite behaviour, correlating strongly with the association score (0.97), but shows low correlation with the detection score (0.58). (iii) HOTA is between these two extremes, correlating reasonably strongly with both detection (0.67) and association (0.94). This explains why in many cases in Tab. 4.2 the HOTA score causes trackers to increase in rank compared to MOTA while simultaneously decreasing in rank compared to IDF1 (and vice versa).

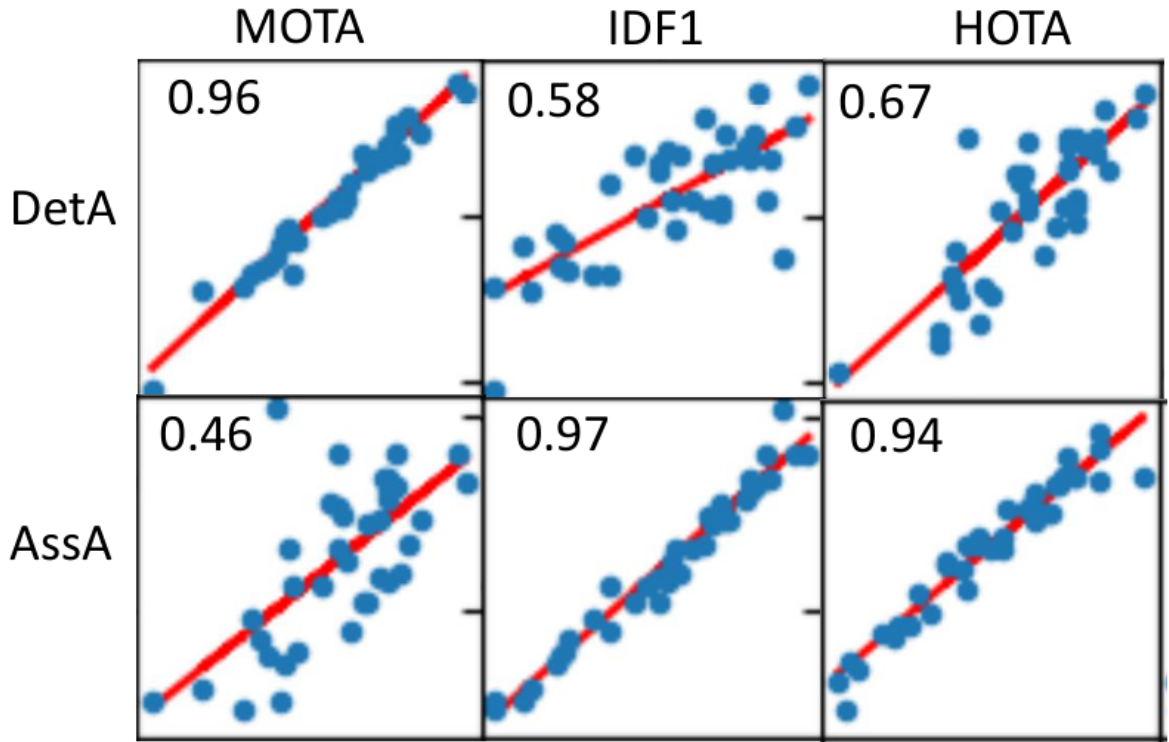


Figure 4.16: Plotting each of the 3 main metrics against both the detection score and the association score for trackers on MOT17. The line of best fit is plotted in red, and the coefficient of determination (R^2) is shown in the top left of each plot.

Note that the correlation is stronger for association than detection. This is not because association is assigned a higher weight by HOTA (both are weighted equally). The reason for this is that there is a wider range of association scores among the trackers, compared to the range of detection scores. This is to be expected, when all trackers are using a set of given public detections as input. Thus the variation in tracking scores is more likely to come from association than detection.

Methods which HOTA ranks higher than MOTA but lower than IDF1 are those for which the association is more accurate than the detection. An example of this is SAS_MOT17 (Maksai & Fua, 2019) which rises 26 places in HOTA compared to MOTA but falls 2 places compared to IDF1. This method specifically focuses on performing accurate association (and they also analyse how IDF1 is better correlated with better association than MOTA), at the cost of detection accuracy. Thus this method performs poorly on MOTA and very well according to IDF1, while performing somewhere in between according to HOTA.

Methods which HOTA ranks higher than IDF1 but lower than MOTA are those for which the detection is more accurate than the association. An example of this is JBNOT (Henschel et al., 2019) which drops 8 places compared to MOTA, but rises 5

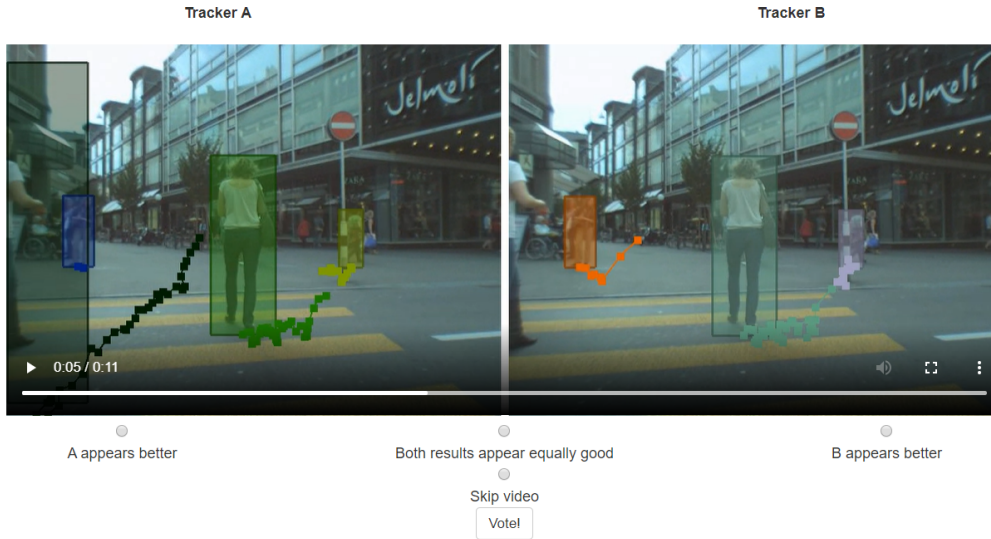


Figure 4.17: An example of user-interface used for the user-experiment and the tracking visualisation.

places w.r.t. to IDF1. This method focuses on improving detection recall, particularly during occlusions by using body joint detectors. However this does not perform association as well as many other methods. Therefore, this method ranks highly according to MOTA, but poorly according to IDF1. HOTA again places it in-between, taking both detection and association evenly into account.

4.11 Human Visual Assessment Study

In previous sections we have provided theoretical analysis as to why HOTA is preferable to other metrics, as well as experimental analysis when using these metrics to evaluate real trackers. In this section we take this analysis one step further and perform a large-scale user-study in order to determine how these metrics align with human judgment of tracking quality. Our study follows many aspects of the design of (Leal-Taixé et al., 2017) which previously attempted to evaluate tracking metrics using human evaluators. This study evaluated MOTA against a number of simple metrics such as Mostly Tracked, Detection Recall and MOTP. We wish to extend their analysis by running a study comparing HOTA, MOTA and IDF1.

Each of the different metrics has its own set of assumptions about what is important for tracking and evaluates against the best practices according to its own assumptions. Here we seek to answer the question of whether the assumptions for each metric align with the assumptions that humans make when viewing objects tracked through a video. While this is not a perfect proxy for the usefulness of these assumptions for any particular tracking application, it is nonetheless useful to know how each of

these sets of assumptions aligns with human ranking. Furthermore, by specifically recruiting MOT researchers to participate in our study, we are able to evaluate how the assumptions of each metric align with the assumptions of the community of people who would be using these metrics. We believe it is a useful property for metrics to evaluate tracking results in a way that is similar to how experts would rank the results when viewing them.

Such an experiment needs to be carefully designed, such that the experiment imposes as little bias to the results as possible. In order to conduct this study, we split the 7 sequences of the test-set of MOT17 up into 6 seconds chunks, which gives us 36 short clips. We performed trial evaluations across video lengths of 3, 6 and 9 seconds and found that the optimal span for this study was 6-second clips. At 9 seconds the video was too long for the human attention span to adequately judge tracking quality accurately. At 3 seconds the videos were too short for the tracking results to be representative of meaningful tracking scenarios. We then evaluated all trackers on the MOT17 benchmark, across these 36 six second video clips, and evaluated the MOTA, IDF1 and HOTA scores. In order to directly compare between metrics in the most user efficient way possible, we designed our experiment as a head-to-head comparison between metrics, such that the user will be presented with a pair of videos showing the tracking results of two different trackers. Each video pair consists of the results for two trackers where two of the metrics significantly disagree on which of the trackers performs better. Users are then to select which tracker performs better, thus agreeing with one of the two metrics in the head-to-head comparison. Users were also given the option to select that both trackers performed equally, or to skip a video pair.

Determining the pairs. We determined which pairs to show by first determining pairs that met our head-to-head requirements, and then by sub-selecting valid pairs based on diversity of sequences, detectors and trackers shown. Two trackers, A and B, are a valid pair to compare for the two metrics M1 and M2 (for a particular sub-sequence and detector input) if they meet the following conditions:

$$S1 \cdot S2 < 0, \quad |S1| > 0.05, \quad |S2| > 0.05 \quad (4.45)$$

where S1 and S2 are defined as:

$$S1 = M1(A) - M1(B), \quad S2 = M2(A) - M2(B) \quad (4.46)$$

The first constraint ensures that the two metrics disagree about which tracker is better (e.g., that the difference between scores for the trackers on one metric should have a different sign than for the other metric). The second two constraints ensure that for both metrics there is a significant difference between the trackers (at least 0.05) so that any difference in ranking between the metrics is significant.

We evaluated 175 trackers on 108 unique combinations of sub-sequence and input detections (36 sub-sequences and three detection inputs). When comparing all pairs

of trackers which met the above constraints there were hundreds of valid pairs per combination of metric. However we were aiming for a smaller set of videos for the user study, as we wished for each pair to be evaluated by a number of different users. We sub-sampled the pairs in such a way that we took at most one pair from each sub-sequence/detection combo and at most one pair that contained each tracker. We did this by greedily taking the pairs which minimise $S1 \times S2$, thus finding pairs for which the metrics maximally disagree, and for each chosen pair removing all pairs which contain the same tracker or the same sub-sequence/detection combination from the pool of valid pairs, and iterating greedily until there is no more pairs to choose from. This gave us 67 pairs for HOTA vs MOTA, 51 pairs for HOTA vs IDF1, and 69 pairs for MOTA vs IDF1.

Result Visualisation. The way tracking results are displayed to users is critically important for such a user study. Depending on how they are displayed, different aspects of tracking could be emphasised. As we have seen in Sec. 4.9 and 4.10, MOTA scores depend more on the quality of detection while IDF1 scores depend more on the quality of association, thus it is important that the visualisation method makes both types of potential errors as obvious as possible to users.

If the visualisation fairly balances the visual saliency of detection and association, then MOTA and IDF1 should perform equally well when compared head-to-head in the user-study, as each is representative of these two different error types. Fig. 4.18 shows that this ends up being the case.

An example of our tracking visualisations (along with the user interface for rating trackers) can be seen in Figure 4.17.

As can be seen we have made detection as obvious as possible by showing bounding boxes with both a thick colored border and a slightly transparent fill. We have also made association as obvious as possible by showing a tracking history of the bottom of each bounding box (in 2D pixel space) as coloured points with lines joining them. This history remains shown for the whole history of a track, and only disappears when that object is no longer present in the current frame. Such a visualisation style allows for both a quick understanding of the properties of trackers, as well as allowing for a conscientious user to take their time and understand all of the complex detections and associations present.

We play videos to users at half the natural frame-rate for easier video clarity. Users are also able to move around frames of the video by either clicking or dragging with the mouse, or by using the arrow keys. Pairs of trackers were shown to users in a random order, and users could evaluate as many pairs as they desired. The videos within each pair were shuffled so that the placement had no impact on which metric was under evaluation.

Results of the User Study. We obtained user study results from 230 participants, 62 of which are multi-object tracking researchers, and 122 of which are computer vision

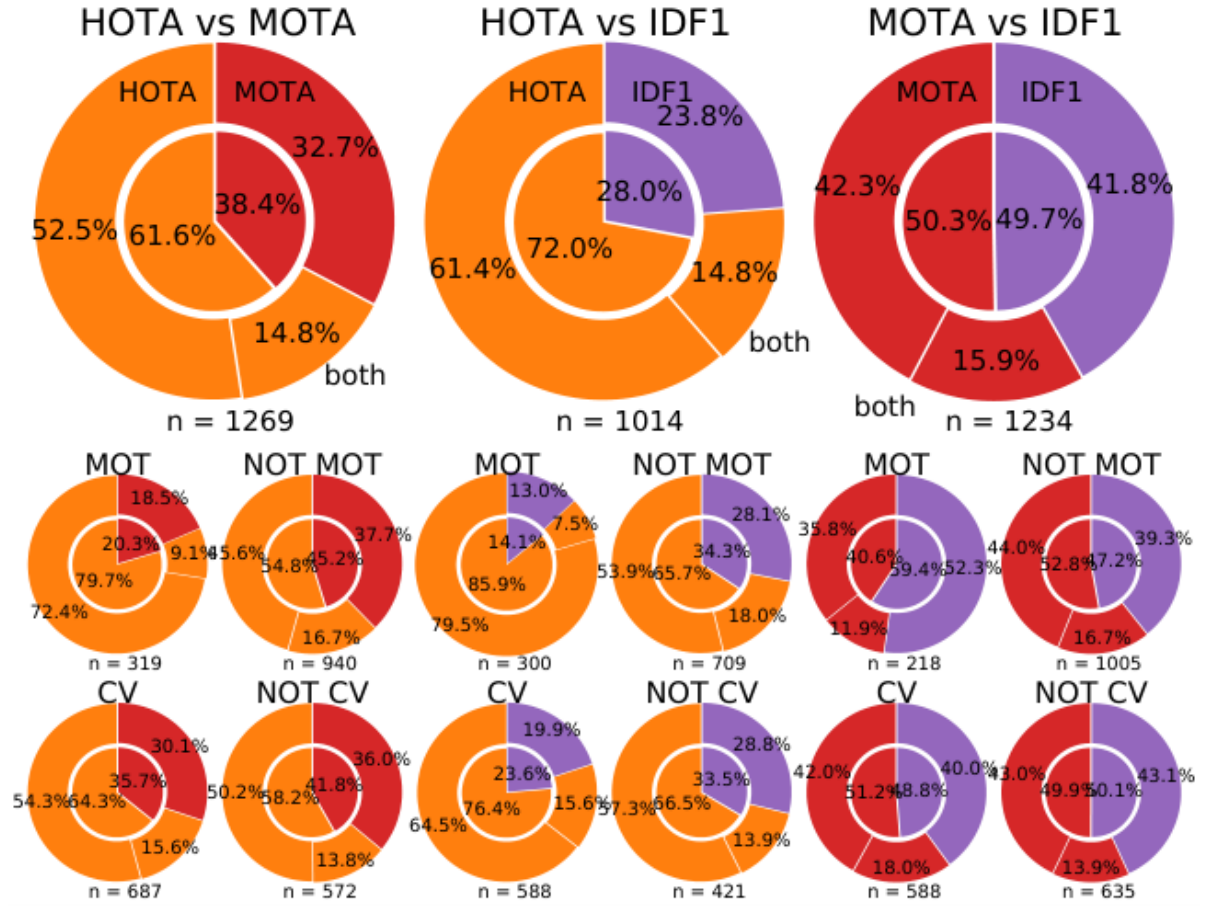


Figure 4.18: Results of our user study showing which metrics better agree with human judgment. Experiments are set up in a head-to-head manner where for a pair of videos the two metrics in the pair disagree on which is better. Users selected the better tracking results, or alternatively rated both the same. *Inner circles are the results excluding those that ranked both equally, while outer circles include these results.* “n=” shows the number of video pairs evaluated for each comparison. Results from all users are shown in the big circles. Results are also shown by user self-reported domain, using information about whether the user is a multi-object tracking researcher (MOT) and whether they are a computer vision researcher (CV).

researchers. On average each user evaluated 9.02 pairs of trackers, for a total of 2075 unique tracker comparisons. On average users took 2 minutes and 13 seconds to evaluate each tracking pair, spending on average 20 minutes evaluating trackers. This is the equivalent of 80 hours spent evaluating tracking results.

A visualisation of results of the head-to-head comparisons between trackers can be seen in Figure 4.18. Note that some pairs were used in multiple metric head-to-heads. As can be seen, looking at all participants, HOTA outperforms MOTA by agreeing with

human evaluators 61.6% of the time compared to 38.4% for MOTA (when excluding those that voted for both). In the comparison of HOTA and IDF1, human evaluators agree even more that HOTA is a better tracking metric, agreeing with HOTA 72.0% of the time compared with only 28.0% for IDF1. In the head-to-head for MOTA vs IDF1 each metric agreed with users around 50% of the time.

For researchers who work in multi-object tracking the levels of agreement with the HOTA metric compared to both alternatives are much higher. Compared to MOTA, users agreed with HOTA 79.3% of the time. Compared to IDF1, users agreed with HOTA 85.9% of the time.

These results show that HOTA better aligns with human judgment of the accuracy of tracking results than previous metrics. The fact that MOT researchers agree even more consistently with HOTA is a strong indication that HOTA is able to successfully evaluate trackers in a way that is relevant for the multi-object tracking community. Evaluating trackers is a difficult task for humans, with often many objects present and extremely complex scenes. The “correct” answer is usually not obvious (as shown by users taking on average more than 2 minutes per pair). However researchers in this field have experience working with such data and know what to look for in good tracking results. As such, the fact that HOTA agrees so strongly with the judgment of MOT researchers is a strong indication of the usefulness for the HOTA metric.

4.12 Discussion

Contributions. In this chapter, we introduce HOTA (Higher Order Tracking Accuracy), a novel metric for evaluating multi-object tracking. Previously used metrics only capture part of what is important for tracking. MOTA is unable to capture association accurately. On the other hand, IDF1 and Track-mAP perform non-intuitively in regards to detection. HOTA tackles these problems with a simple, elegant formulation that equally weights detection and association accuracy.

We argue analytically and experimentally why our proposed metric is preferable over the alternatives, testing HOTA using state-of-the-art trackers on the *MOTChallenge* benchmark. Furthermore, we perform a large-scale user study and demonstrate that human visual assessment of tracking accuracy aligns better with HOTA compared to both MOTA and IDF1.

HOTA has changed the nature of tracking research, laying the groundwork for new algorithms to be designed and benchmarked against a metric that measures both detection and association quality.

Connection to other Chapters. This project was originally conceived, after working on the project presented in Chapter 2, and realising that the evaluation metrics everyone (including myself) were using for evaluating multi-object tracking were entirely sub-par and actually actively harming progress in the tracking community. After finishing

the work in this chapter, one of the obvious remaining limitations in the evaluation of multi-object tracking was that it was limited to evaluating only a few pre-selected object classes. This limitation was then addressed in the work in the following Chapter 5.

Influence. The work in this chapter has had a massive influence since it was published in IJCV in 2021. It has become the default evaluation metric for evaluating multi-object tracking methods, and used as the default for both of the biggest tracking benchmarks: MOTChallenge and KITTI. It has been cited in over 850 academic papers which have used this metric to evaluate their tracking methods, and used the insights gleaned from using this metric to improve their methods toward actually being more performant and useful in the real world. The code-base for this project has been starred over 950 times on GitHub. This project has successfully become one of the enduring core contributions to tracking literature.

Opening up Open-World Tracking

The ability to track and detect any object, even those not seen before during training, is very important for self-driving systems. If these systems cannot recognize new objects, they can be unsafe when used in real life. Unfortunately, most current systems have this problem. One big challenge in improving object tracking is that it is hard to measure how well different methods work. There is a need for a standard way to compare different approaches to help this research area grow.

In this chapter, we present a new standard for testing how well systems can detect and track both known and unknown objects in a wide range of situations. We introduce TAO-OW: Tracking Any Object in an Open World. This new benchmark helps us understand where current multi-object tracking methods stand and sets a starting point for this task. It also points out what the future challenges might be.¹

This chapter is based on our paper ([Luiten* et al., 2022](#)), published as an Oral Presentation in the Conference for Computer Vision and Pattern Recognition (CVPR) in 2022.

5.1 Introduction

Familiar scenarios are straightforward to understand. Vision systems, after being fed millions of images of cars and pedestrians, are fairly reliable in detecting these objects, determining their identities and locations, and tracking their movement across time.

*New and unseen scenarios pose a **significant** challenge.* For instance, how would an autonomous vehicle react to a plane making an emergency landing on the road ahead, or to a novel children’s toy suddenly appearing on the roadway? The ability

¹This paper was originally conceived, organised and written with equal contribution together by Aljosa Osep, Achal Dave and myself. We brought on Yang Liu early in the project in order to run baseline methods, and later on in the project brought in Idil Esen Zulfikar to help him with this. Deva Ramanan, Bastian Leibe and Laura Leal-Taixe contributed discussion and advice along the way.

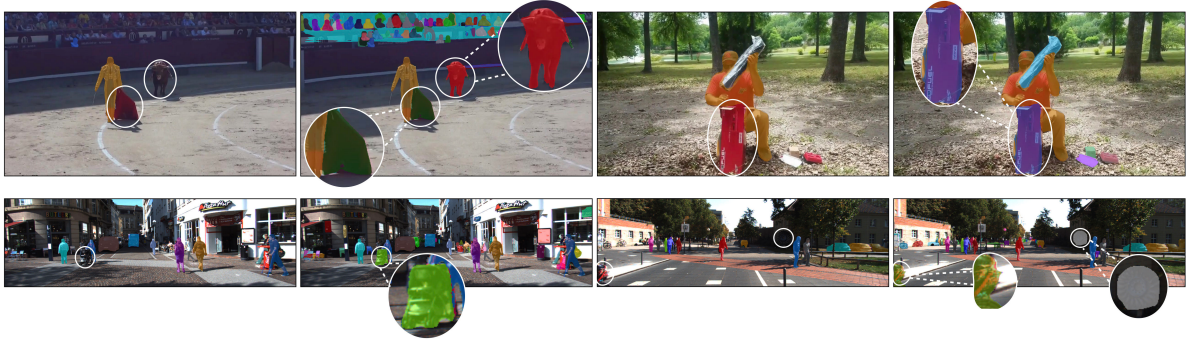


Figure 5.1: *Each pair left:* The prevalent methodology in multi-object tracking engages in the detection, tracking, and sometimes segmentation of objects within certain, pre-established semantic categories, such as vehicles and pedestrians (Voigtlaender et al., 2019b). *Each pair right:* The output from our fundamental tracking system is adept at tracking items, for instance, a child’s stroller, which were not identified in the training dataset of the model. The noteworthy contribution of this section is the debut of the inaugural benchmark tailored to assess the capabilities of object tracking systems in open-world scenarios. These scenarios provide trackers with limited insights into the visual realm, accepting the premise that *training object detectors to recognize every potential semantic class is an unfeasible endeavor*.

of current vision systems to adapt to such unexpected and unfamiliar situations is questionable. Will these systems, designed to detect and track potential hazards, be able to recognize these new elements at all? Or could they fail to do so, with potentially tragic consequences, such as overlooking the *child stroller* shown in Fig. 5.1, bottom-left?

Methods for tracking and detection are quite effective for objects that are extensively documented in data collections. However, unless we develop systems capable of handling objects they have never encountered before, vision systems will remain insufficiently safe for deployment in real-world scenarios. Furthermore, the approach of simply gathering more data is not scalable when considering the endless diversity of potential unknown occurrences.

There are numerous instances suggesting that existing vision systems falter in the face of novel situations (Petrović et al., 2020), yet we lack the quantitative means to measure this deficiency, nor can we gauge improvement, owing to the absence of benchmarks for assessment.

In this chapter, we introduce a novel benchmark (**TAO-OW: Tracking Any Object in an Open World**) that evaluates detection and tracking in an open-world context. Traditional benchmarks for closed-world multi-object tracking (Geiger et al., 2012; Dave et al., 2020; Dendorfer et al., 2020b) and their associated methods (Leibe et al., 2008b; Bergmann et al., 2019; Voigtlaender et al., 2019b) concentrate on monitoring a specific collection of object classes that are commonly encountered. Conversely, our Open-World Tracking (OWT) challenge mandates the tracking of *all* objects, with an

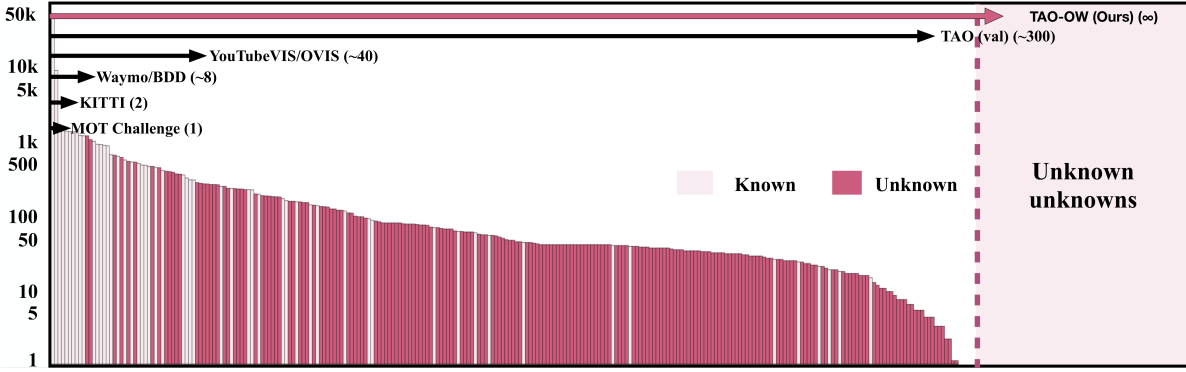


Figure 5.2: TAO-OW Benchmark class distribution within the validation subset, depicting *known* classes that are provided with training data, alongside *unknown* classes that act as representatives for the endless variety of objects that could emerge in an open-world scenario (*unknown unknowns*). Observe that the y-axis is on a **logarithmic scale**.

emphasis on the system’s ability to monitor classes it has not been trained to recognize (*unknown* objects) as well as those included in the training dataset (*known* objects).

Evaluating performance in an Open-World setting presents considerable challenges. It is necessary to define a restricted set of *known* objects for algorithm training, which ought to be sufficiently varied and encompassing to represent a realistic subset of objects for which data might typically be gathered. At the same time, testing must occur on a substantial number of *unknown* objects—those not shown as labeled examples during training.

We build this benchmark upon the TAO dataset (Dave et al., 2020), a broad repository of video material sourced from a variety of contexts such as driving, cinematic content, and day-to-day activities. This diversity is critical for encompassing a broad spectrum of *unknown* objects. For *known* classes, we utilize the 80 classes from the COCO dataset (Lin et al., 2014), representing a broad array of commonplace objects. This leaves in excess of 700 *unknown* object categories for assessing how well algorithms perform on objects beyond their training scope. Figure 5.2 illustrates our TAO-OW benchmark, highlighting its intrinsic long-tail distribution of object classes, the delineation between *known* and *unknown* categories, and draws a comparative analysis with past tracking benchmarks (Geiger et al., 2012; Xu et al., 2018; Dendorfer et al., 2020b; Sun et al., 2020; Yu et al., 2020; Qi et al., 2021), all of which are confined to closed-world assessments over a limited selection of object categories.

One of the fundamental challenges in open-world evaluation is the impracticality of exhaustively annotating every single object that needs to be detected and tracked. By the very nature of the task, we aim not to penalize trackers for identifying and following unknown, unannotated objects. To address this dilemma, we introduce a novel evaluation metric titled Open-World Tracking Accuracy (OWTA). This metric

effectively separates the assessment of detection and tracking, enabling accurate tracking performance measurement in situations where trackers are not penalized for detecting additional, unannotated objects. Such evaluations are made feasible by enforcing a requirement that the proposed objects are presented as distinct, non-overlapping segmentation masks.

Equipped with our Open-World Tracking benchmark and evaluation metrics, we scrutinize a number of methods that have attempted open-world tracking before there was a unified evaluation framework (Ošep et al., 2018; Dave et al., 2019; Luiten et al., 2020). A pivotal element of this chapter is the comprehensive examination of an array of methodologies. From this scrutiny emerges our proposition of an open-world tracking methodology which not only exhibits the best performance on our Open-World Tracking Benchmark but also shows strong competitiveness on earlier closed-world benchmarks. This is particularly noteworthy given that our method was neither specifically conceived nor refined for these previous benchmarks.

The **core contribution** of this research is to forge a novel path in vision-based multi-object tracking, advancing beyond the confines of existing closed-world benchmarks. We define the concept of Open-World Tracking by (i) introducing a benchmark equipped with an apt recall-oriented evaluation metric to measure progress, (ii) reviewing current design frameworks, offering a comprehensive suite of baselines derived from leading methods in the closed-world domain, and (iii) detailing a robust technique that is effective in both open- and closed-world tracking scenarios. Our empirical findings indicate that detectors from the closed-world are quite adept at *localizing* even those objects they have not seen before. Nevertheless, the *tracking* of unknown objects proves to be a more formidable task than that of familiar objects.

5.2 Related Work

Related tasks and benchmarks. The field of multi-object tracking (MOT) presents the complex challenge of pinpointing objects spatially and temporally, often amidst the complexities of densely populated settings. Contemporary MOT datasets are predominantly tailored to closed-set tracking within video streams (Geiger et al., 2012; Yang et al., 2019; Dendorfer et al., 2020b; Wen et al., 2020) or LiDAR data (Caesar et al., 2020; Sun et al., 2020). There has been a shift in recent research towards achieving pixel-level accuracy in segmenting tracked objects across video frames (Milan et al., 2015; Voigtlaender et al., 2019b; Yang et al., 2019; Kim et al., 2020; Weber et al., 2021) or LiDAR sequences (Aygün et al., 2021), with a focus on performance among the less common object classes (Dave et al., 2020). More aligned with our research are the domains of unsupervised video object segmentation (UVOS) (Caelles et al., 2019) and motion segmentation (Irani & Anandan, 1998; Shi & Malik, 1998; Bideau & Learned-Miller, 2016), which require the identification and delineation of various objects throughout a video sequence, especially those exhibiting significant movement. Yet, it is important

to note that the majority of classes in these benchmarks overlap with those in COCO, and most methods (Athar et al., 2020; Luiten et al., 2020) gain impressive results by leveraging COCO for training. Contrarily, our investigation deliberately targets classes that extend *beyond* the scope of COCO.

Multi-object tracking. Initial approaches in vision-based tracking (Wren et al., 1997; Haritaoglu et al., 2000; Paragios & Deriche, 2000) and in the field of robotic perception (Teichman et al., 2011; Moosmann & Stiller, 2013) leveraged class-independent, bottom-up segmentation techniques as a primary tracking signal, e.g., using clustering of LiDAR point clouds (Moosmann et al., 2009; Wang et al., 2012) or the segmentation of foreground from background (Jain & Nagel, 1979; Wren et al., 1997; Stauffer & Grimson, 2000). Advances in vision-based MOT were largely driven by the *tracking-by-detection* approach, dependent on detectors pre-trained to recognize objects. Initial efforts were concentrated on enhancing data association methods (Reid, 1979; Leibe et al., 2008a; Brendel et al., 2011; Leal-Taixé et al., 2011; Pirsiavash et al., 2011; Milan et al., 2016b) and meticulously crafting cues for appearance (Mitzel et al., 2010; Geiger et al., 2014; Milan et al., 2014) and motion (Leal-Taixé et al., 2014; Choi, 2015). Subsequent endeavors have shifted towards a data-driven methodology, focusing on developing robust appearance models (Kim et al., 2015; Leal-Taixé et al., 2016), learning target regression strategies (Bergmann et al., 2019) and enhancing detection associations with the help of graph neural networks (Brasó & Leal-Taixé, 2020). Most of the progress in the domain of closed-set multi-object tracking can be ascribed to the introduction of new datasets, benchmarks, and evaluation protocols. Despite this, the assessment of MOT remains confined to controlled, closed-set environments where the categories of objects are predetermined and included within the training datasets.

Beyond closed-world tracking. The *tracking-by-detection* paradigm has been expanded to encompass generic objects (Ošep et al., 2018; Dave et al., 2019; Ošep et al., 2020) as well as unsupervised video object segmentation (UVOS) (Dave et al., 2019; Xie et al., 2019; Luiten et al., 2020), utilizing proposal mechanisms trained without category-specific labels (Pinheiro et al., 2016; He et al., 2017). Despite these developments, the absence of a relevant evaluation framework for open-world scenarios rendered it uncertain how these methodologies would perform with arbitrary object types.

In concurrent **parallel work** (Wang et al., 2021), the focus has been on annotating diverse classes within the human-focused Kinetics400 dataset (Kay et al., 2017). This research emphasizes data curation and advocates the use of the extant closed-world Track mAP metric (Yang et al., 2019) for assessment purposes. However, this metric has been subject to significant critique (Luiten et al., 2021) for its challenging interpretability, insensitivity, and inability to differentiate between types of errors—issues that are particularly acute in the open-world tracking evaluation. Additionally, this metric presupposes comprehensive labeling of all detectable objects, which is impractical. Moreover, the dataset is circumscribed to human-related actions. Our work diverges

by exploring open-world tracking within a broader variety of contexts, encompassing footage from several distinct environments. This breadth is indispensable for probing open-world challenges, reducing bias and fostering greater generalizability (e.g., failure on objects not placed centrally within the frame). Lastly, our work scrutinizes previous research on open-world tracking, extracting fundamental components of these methods to undertake an exhaustive analysis and to formulate a novel baseline. This baseline demonstrates proficiency under both open- and closed-world circumstances.

Open-set recognition, detection and segmentation. The domain of open-set recognition is aimed at reducing misclassification between *known* classes, which are familiar during training, and *unknown* classes, which might emerge in uncontrolled environments (Scheirer et al., 2012; Jain et al., 2014; Scheirer et al., 2014; Bendale & Boulton, 2016). Recent work in object detection also ventures into these open-set conditions (Dhamija et al., 2018; Miller et al., 2018). In contrast, open-world recognition, as delineated by (Bendale & Boulton, 2015; Liu et al., 2019b), obliges models to explicitly identify *unknown* instances unseen in training and refine detectors to acknowledge these newfound instances. This concept has been applied within automotive contexts (Osep et al., 2018), with detectors adapting via retraining on clusters of *unknown* object trajectories discovered in video streams (Ošep et al., 2018, 2019). In parallel, (Hwang et al., 2021) showcased a method for detecting *unknown* objects by generating and grouping proposals from unannotated (*void*) image regions, then utilizing these groupings as quasi-labels for further training.

Joseph et al. (Joseph et al., 2021) introduce an augmentation to the Faster R-CNN framework (Ren et al., 2015) designed to discern between known and unknown classes. This is achieved by incorporating a contrastive objective, which amplifies the separation between *known* and *unknown* objects within the feature space. Our approach, however, does not concentrate on diminishing confusion between distinct semantic categories or on progressive learning strategies. Our focus is on proficiently identifying and following objects across both *known* and *unknown* classes, without the necessity for semantic classification of the tracked entities. We propose that the task of tracking any object is a fundamental challenge that should be prioritized over recognition. We view our research as laying the groundwork for the application of such methodologies to video streams observed by intelligent systems.

5.3 Opening up Open World Tracking

Current trackers predominantly function within the scope of specific, predefined object categories, such as humans or vehicles, which are identified and tagged within their training datasets—these we classify as *known objects*. Our objective is to extend this evaluation to encompass *unknown objects* as well, which are those not demarcated in the training data. An open-world tracker’s role is to discern and monitor *all*



Figure 5.3: Illustrating the concept of **Unknown Unknowns**, this figure presents instances of objects not included in the TAO (Dave et al., 2020) lexicon, yet successfully tracked by our algorithm.

objects—regardless of their classification as *known* or *unknown*—within a video feed. Assessing the effectiveness of trackers under these conditions is inherently complex. For one, exhaustively labeling every object within a video is a task with prohibitive costs. Real-world datasets seldom catalog all present objects due to the high expenses involved, usually constraining themselves to notating a select number of classes (e.g., KITTI (Geiger et al., 2012), MOTChallenge (Dendorfer et al., 2020b)), or specific instances (e.g., TAO (Dave et al., 2020)). Moreover, establishing a universal yet coherent definition of an *object* is an intricate challenge (Alexe et al., 2012).

In an attempt to overcome these hurdles, we have adopted a *recall*-focused approach for our evaluation, drawing inspiration from the initial ventures into the assessment of object proposal methods (Endres & Hoiem, 2010; Alexe et al., 2012) and its subsequent applications in the realms of zero-shot object detection (Bansal et al., 2018) and open-world LiDAR segmentation (Wong et al., 2020). While arriving at an explicit definition of an *object* may pose a significant challenge, there exists a consensus among people regarding the fundamental perception of objects, enabling them to label arbitrary objects within a given environment (Gupta et al., 2019). Thus, we derive our positive object instances from the convergence of multiple human annotators on whether *something is an object*. This consensus forms the basis of our ground truth, allowing us to evaluate the extent to which a tracker can *recall* these agreed-upon object instances.

Defining the notion of a false positive (FP) is non-trivial as we can only expect a *subset* of objects to be labeled. If we consider unlabeled regions as non-objects (FPs), we may be penalizing the tracking system for tracking regions that could still be considered to be valid objects. See Fig. 5.3 for an example of objects not labeled in the TAO (Dave et al., 2020) dataset, but correctly tracked by our baseline tracker.

Open-World Tracking Accuracy (OWTA). We introduce OWTA (Open-World Tracking Accuracy), an extension of the HOTA metric (Luiten et al., 2021) designed for the evaluation of multi-object tracking in an open-world scenario. OWTA incorporates two fundamental components:

- **Association Accuracy (AssA):** This term quantifies the accuracy with which the tracker can maintain consistent identities of objects across frames.
- **Detection Recall (DetRe):** This term measures the tracker’s ability to locate and identify all relevant objects within the video footage, regardless of their prior exposure in the training data.

Both AssA and DetRe are assessed against a variable localization threshold α , which defines the degree of spatial precision required for a detection to be considered correct. OWTA is then computed by integrating the product of AssA and DetRe across all values of α , offering a comprehensive measure of a tracker’s performance in open-world conditions.

$$\text{OWTA}_\alpha = \sqrt{\text{DetRe}_\alpha \cdot \text{AssA}_\alpha}, \quad \text{DetRe}_\alpha = \frac{|\text{TP}_\alpha|}{|\text{TP}_\alpha| + |\text{FN}_\alpha|}.$$

Recall-focused Evaluation for Open-World Tracking. In open-world tracking, where the objective is to detect and track as many objects as possible, including those that are not previously labeled in the training dataset, the emphasis is on recall rather than precision. To reflect this goal, the *Detection Recall (DetRe)* aspect of our OWTA metric deliberately does not penalize trackers for false positives.

This approach aligns with the philosophy of open-world recognition, which aims to embrace the vast variety of objects that can appear in an unconstrained environment. It is underpinned by research in zero-shot object detection and open-world recognition tasks where the detection of previously unseen classes is crucial (Bansal et al., 2018; Wong et al., 2020).

By focusing on recall, we encourage the development of tracking systems that can generalize beyond the limited scope of pre-defined classes, facilitating the identification and tracking of a broader array of objects without the constraint of exhaustive annotation. This method allows for a more inclusive and extensive assessment of a tracker’s detection capabilities in the open world.

Association Accuracy in Open-World Tracking. The *Association Accuracy (AssA)* component of our OWTA metric quantitatively measures the consistency of a tracker in maintaining correct track identities over time. Introduced in the closed-world tracking context (Luiten et al., 2021), it has been adapted to assess the performance in the more challenging open-world scenario.

For each instance of overlap between a predicted track p_t and a true ground truth track g_t , the following values are computed:

- **True Positive Associations (TPA):** Counts instances where p_t and g_t both agree on the detected object's presence in a given frame.
- **False Positive Associations (FPA):** Occurs when p_t claims an object's presence in a frame where g_t does not, indicating an over-prediction.
- **False Negative Associations (FNA):** Arises when g_t has an object that p_t fails to detect in a given frame, signifying an under-prediction.

$AssA$ is then calculated using the intersection-over-union (IoU) principle applied to the TPA, FPA, and FNA, reflecting the accuracy of association:

$$AssA_{\alpha} = \frac{1}{|TP_{\alpha}|} \sum_{c \in TP_{\alpha}} \frac{TPA_{\alpha}(c)}{TPA_{\alpha}(c) + FPA_{\alpha}(c) + FNA_{\alpha}(c)}.$$

This metric effectively captures the tracker's ability to associate detections correctly across frames, which is a critical aspect of maintaining coherent tracks throughout a video sequence. By averaging over all true positives, $AssA$ provides a comprehensive assessment of the tracker's performance in sustaining object identities reliably over time, an essential feature for practical applications in dynamic and unpredictable open-world environments.

The *Association Accuracy* ($AssA$) term is particularly beneficial for the evaluation of tracking in the open-world scenario because it is inherently class-agnostic; it focuses on the tracker's ability to maintain identity consistency over time, irrespective of the object class. This attribute makes it suitable for scenarios where objects may not be exhaustively labeled or where classes are unknown or diverse.

In the context of $AssA$, False Positive Associations (FPA) are considered only when there is an existing match to a ground truth track. This means FPAs are computed in situations where the tracker has identified an object (a potential false positive), but it has been associated with a true positive detection. FP tracks that do not have any match with ground truth data do not contribute to FPA and, consequently, do not impact $AssA$ directly. This is a significant advantage over metrics that require a full account of all detections, such as Track mAP (Yang et al., 2019) and IDF1 (Ristani et al., 2016).

Track mAP and IDF1 require a comprehensive accounting of all detections, true and false, to compute the final metric. This typically necessitates a densely labeled dataset to provide ground truth for all possible detections. In contrast, $AssA$ can function effectively even in sparsely labeled datasets common in open-world settings, where exhaustively labeling every instance is impractical.

This factorization of the evaluation metric—separating the ability to track consistently from the need to recognize every object—allows for a more scalable and practical

approach to evaluate trackers in diverse and less-controlled environments. It helps to shift the focus from recognizing every possible object to the more fundamental task of maintaining consistent and reliable tracks over time, which is crucial for many real-world applications where object classification can be secondary or can be addressed subsequently.

During the evaluation phase, the requirement is for methods to produce non-overlapping masks that represent tracks, ensuring each pixel in a frame is designated exclusively to a single track or designated as background. To attain a high level of recall, a method must accurately assemble and track pixel groups over time. The possibility of a naive solution that creates boundlessly many tracks is precluded, as the initiation of a track negates the presence of any other track within the same pixel space. This constraint harmonizes our Open-World Tracking (OWT) endeavor with the prevailing direction in tracking research, which prioritizes precise object tracking at the pixel level, thus shifting away from the traditional bounding-box approach. Our task parallels the Open-World variant of Multi-Object Tracking and Segmentation (MOTS) (Voigtlaender et al., 2019b) or Video Instance Segmentation (VIS) (Yang et al., 2019).

5.4 TAO-OW Benchmark

The establishment of a definitive and dependable benchmark is essential to facilitate advancement in the field. Hence, we introduce the Tracking Any Object in an Open World (TAO-OW) benchmark.

Dataset. Contrasting with the scope of most existing MOT benchmarks such as (Voigtlaender et al., 2019b; Yang et al., 2019; Sun et al., 2020; Yu et al., 2020; Qi et al., 2021), the TAO dataset (Dave et al., 2020) unveiled recently encompasses an extensive array of object classes. TAO amasses close to 3,000 video entries (including 593 training, 988 validation, and 1,419 test segments), aggregating over 100,000 frames annotated and an expansive categorization spanning 800 object classes. Crucially, TAO’s annotations are not confined by predefined object classes; the annotators were instructed to mark *any moving object within the footage*. This method gives rise to a long-tailed class distribution (refer to Fig. 5.2), effectively mirroring the vast assortment of objects that might be encountered in real-world environments. The capability to construct trackers adept at following every object within this vast video repository would potentiate their applicability across a myriad of uncontrolled and *open-world* contexts.

In its standard form, the TAO dataset is structured for a *closed-world* scenario wherein all object classes are specified and exemplified during the training phase. We have reconfigured this dataset to suit an open-world paradigm by deliberately excluding certain classes from the training process, albeit maintaining them within the evaluation scope. Additionally, we extend the evaluation to encompass an extra 143 classes that

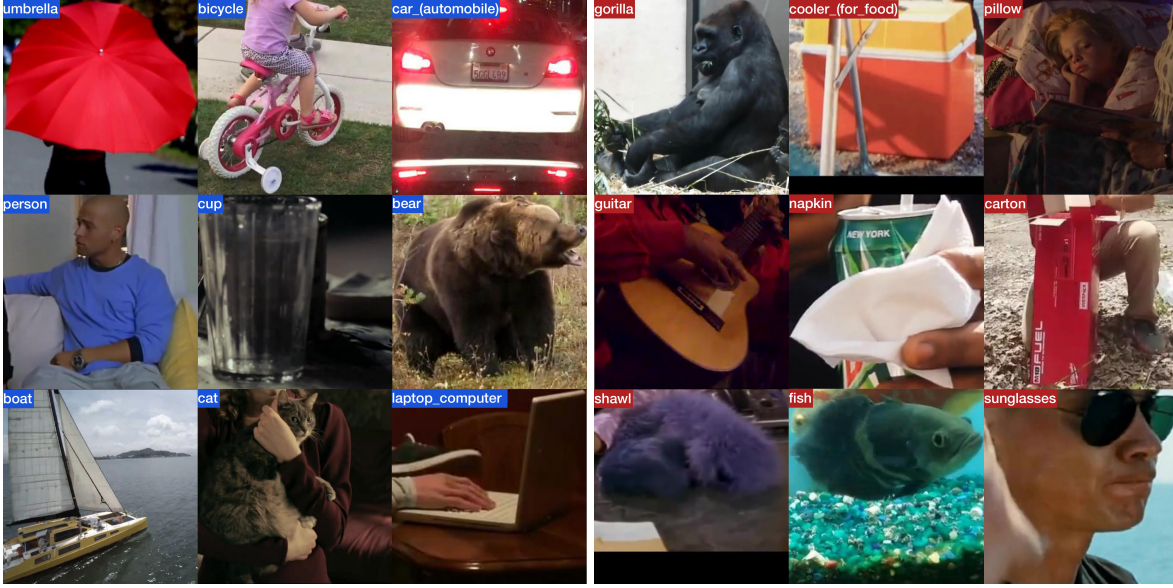


Figure 5.4: Examples of *known* object categories (left) and *unknown* object categories (right).

are exclusive to the test set, bypassing the validation set. These classes are denoted as *unknown unknowns*. Such a setup facilitates an assessment within open-world conditions, particularly for classes that were not incorporated during the model’s parameter validation phase.

Delineating the *known* and *unknown* Categories. In the course of delineating the *known* and *unknown* categories for our evaluation, we take into account several critical considerations: firstly (i), we ensure there is a sufficiently large and diverse dataset encompassing the *known* categories to facilitate the training of robust models that can generalize to a broader array of categories; secondly (ii), we maintain a substantial quantity of *unknown* categories to enable a comprehensive assessment of the tracking performance for these classes; and thirdly (iii), the chosen *known* categories should encompass those typically involved in closed-world multi-object tracking (MOT), enabling a seamless transition to assess the performance of closed-world trackers within an open-world context. Consequently, we designate the categories from the COCO dataset as *known*, which includes 80 prevalent categories such as people, animals, vehicles, portable items, and home appliances.

Composition of the *known* and *unknown* Sets in TAO Validation. The TAO validation dataset incorporates 52 out of the 80 COCO classes, resulting in a total of 87,358 individual object trajectories that we classify as the *known* set. In addition to these, the TAO validation dataset features a further 209 distinct classes that do not coincide with any COCO categories, comprising 20,522 unique object trajectories. Within

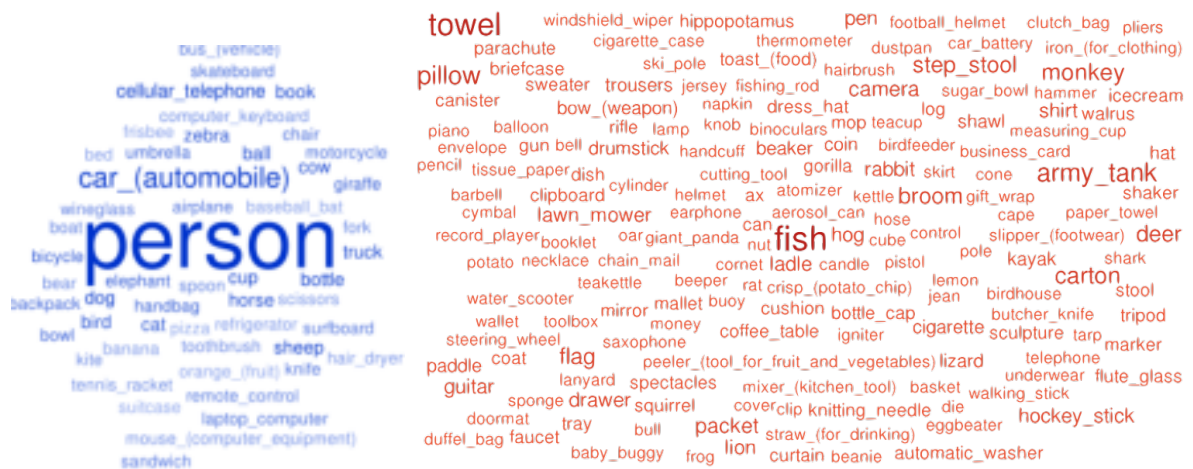


Figure 5.5: TAO-OW classes. Word cloud showing *known* (left) and *unknown* (right) classes in our TAO-OW benchmark, with word-size proportional to frequency.

these *unknown* classes, which are absent from the COCO dataset, the most frequently occurring are *fish*, *towel*, and *pillow*, with 1,274, 1,128, and 688 tracks respectively. The *unknown* set is replete with diverse and intriguing classes that merit tracking, with some favorites among the authors being *walrus*, *ice cream*, *drum*, *frog*, *gift wrap*, and *binoculars*. Illustrative examples of videos featuring both *known* and *unknown* objects are displayed in Fig. 5.4. Moreover, Fig. 5.5 provides a word cloud visualization for all the *known* and *unknown* classes within the TAO-OW validation set, with the size of each word corresponding to the count of annotated trajectories for each class. To circumvent any potential bias in the evaluation process towards classes that bear similarity to the *known* ones, we distinguish 41 classes as “distractors” following the precedent set in closed-world tracking benchmarks. These classes, which are excluded from the evaluation, include specific instances like *cab* (a subtype of *car*) and *water bottle* (a subtype of *bottle*).

Dense Labeling and Evaluation Methodology in TAO. Despite TAO’s breadth, it is not exhaustively labeled; numerous objects within its videos do not have annotations. In traditional *closed-set* tracking scenarios, this could result in penalization for trackers that identify these correctly but unannotated objects. Nevertheless, for Open-World Tracking (OWT), the impact is mitigated as the evaluation is grounded in a recall-centric metric—Open-World Tracking Accuracy (OWTA)—outlined in Sec. 5.3. Additionally, TAO annotates objects using bounding boxes rather than segmentation masks. However, OWT necessitates mask outputs from tracking methods. To reconcile this during evaluation, ground truth bounding boxes, which are non-amodal and only encapsulate the visible portions of objects, can be used as a basis to assess masks by converting the masks back into bounding boxes.

TAO-OW dataset split. In crafting the TAO-OW dataset, we divide it into training, validation, and testing subsets, all derived from the original TAO compilation. The training subset is curated to include only *known* class annotations, deliberately omitting any annotations outside this category. The validation subset encompasses all object instances from TAO, duly categorized into *known* or *unknown*, contingent on their concordance with COCO-defined classes. As for the testing subset, it incorporates the entire spectrum of objects, labeling those recognized in the validation subset as *known* or *unknown*, and demarcating the additional classes as *unknown unknowns*.

This delineation is strategic, especially for the *unknown unknowns* in the test set, ensuring a stringent “hold out” criterion. Classes featured in the test set are deliberately excluded from the validation set to preserve the integrity of a true testing scenario. The presence of *unknown unknown* classes is crucial; it is the litmus test for the model’s adaptability and robustness in an authentic open-world environment, transcending the confines of the classes defined in the training and validation subsets.

5.5 Designing Open-World Trackers

Establishing Baselines for TAO-OW. In the realm of benchmarks, baselines act as foundational references, and the creation of robust baselines is indispensable. For TAO-OW, the methods most relevant to our context (Ošep et al., 2018; Dave et al., 2019; Luiten et al., 2020) are not directly transferable to our domain: (Ošep et al., 2018) is contingent on the availability of stereo videos; (Dave et al., 2019) operates under the premise that objects are in motion; and (Luiten et al., 2020) presupposes the continuous visibility of objects across all frames.

Given these constraints, our endeavor extends beyond mere adaptation; it encompasses a critical dissection and synthesis of the core tenets these methods embody. This chapter, therefore, not only analyzes but also integrates these principles to formulate a comprehensive framework tailored for open-world tracking—a framework that serves as a cornerstone for future explorations and evaluations in this space.

Baseline Development for OWT via Tracking-by-Detection. To construct a robust baseline for Open-World Tracking (OWT), we scrutinize the prevalent tracking-by-detection (TBD) paradigm that has long steered the field of Multi-Object Tracking (MOT) (Dendorfer et al., 2020b). Our goal is to understand and reconfigure TBD for OWT. We discern that the standard TBD process can be stratified into four sequential stages (depicted in Fig. 5.6):

1. *Object Proposal Generation:* Initial detection of potential objects in individual images.
2. *Short-term Proposal Association:* Calculation of cross-frame proposal similarities, which is fundamental for data linkage.

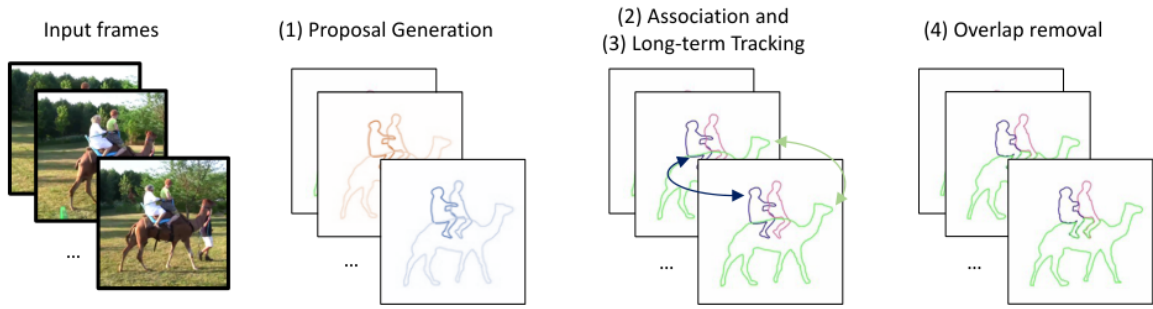


Figure 5.6: The formulation of our Open-World Tracking Baseline (OWTB) is influenced by the tracking-by-detection paradigm. Our approach involves (1) extracting object proposals, (2) calculating association scores across frames, which inform (3) the creation and maintenance of object tracks, and ultimately, (4) reconciling any spatial-temporal overlaps to prevent conflicts between concurrent tracks.

3. *Track Management*: Application of similarity metrics to correlate proposals and maintain track continuity.
4. *Pixel-to-Track Allocation*: Final determination of exclusive track assignments for each pixel.

We delve into each of these stages methodically, opting for the most effective techniques at each juncture as the foundation for subsequent phases. This strategic approach enables us to navigate and streamline the otherwise vast and complex space of design possibilities.

5.5.1 Proposal Generation (1)

In line with the *tracking-by-detection* approach, our initial step is to gather evidence of potential objects at the image level. Leveraging insights from previous studies (Ošep et al., 2018; Dave et al., 2019; Dhamija et al., 2020), we employ learned object proposal mechanisms such as the Region Proposal Network (RPN) (Pinheiro et al., 2015; Ren et al., 2015; Pinheiro et al., 2016). These mechanisms are adept at differentiating object-centric regions from the *background*, enabling them to recognize object classes not included in the training set, as evidenced by (Pinheiro et al., 2015, 2016). Our analysis utilizes Mask R-CNN (He et al., 2017), investigating its efficacy in identifying *unknown* objects. We train our model on the 80 recognized classes and then appraise its capacity to handle TAO-OW’s *known* and *unknown* classes distinctly. To assess the detector’s utility in generating proposals, we apply a low score threshold to gather the top 1000 proposals produced by the model.

Proposal Recall Analysis. The capacity of the object proposal mechanism to include objects of interest within its top 1000 predictions is crucial for the success of our tracking

	Overall	Small size	Medium size	Large size
<i>known</i> / <i>unknown</i>	95.4/75.5	91.4/66.1	98.4/85.9	99.7/98.2

Table 5.1: Recall Analysis Across Object Sizes for 1k Proposals per Image. This figure elucidates the object detection recall across different object sizes, considering a large sample of object proposals (1000 per image). The results indicate that the models exhibit a strong performance when identifying *known* objects and *large unknown* objects. However, there is a notable decline in recall for *small unknown* objects, suggesting a challenge for the model to detect less prominent elements that do not belong to the classes seen during training.

system. Table 5.1 delineates the recall performance for both *known* and *unknown* objects, categorized by their size: Large (with an area ratio ≥ 0.3), Medium (area ratio between 0.03 and 0.3), and Small (area ratio < 0.03). These categories are relative to the overall image size. Remarkably, the model exhibits robust recall for large objects, irrespective of their known or unknown status. However, it falters significantly when it comes to small *unknown* objects, indicating a generalization gap in the model’s ability to detect smaller, less conspicuous objects from the *unknown* classes.

Considering the impracticality of employing all 1000 proposals for tracking, we focus on enhancing the model’s proficiency at distinguishing *unknown* objects from the pervasive background noise. As depicted in Figure 5.7 (left), we analyze detection recall against the number of proposals for various scoring heuristics and also present the area under the curve (AUC) for each.

Notably, Figure 5.7 demonstrates that the highest confidence score for the *known* class predictions (*score*) does not serve as a reliable indicator, achieving 0.89 AUC for *known* and merely 0.59 AUC for *unknown* objects. In contrast, the objectness score (*obj.*) computed by the RPN stands out as a more effective metric, scoring 0.92 AUC for *known* and 0.67 AUC for *unknown* objects. The score for background classification (*bg*), defined as the likelihood for none of the classes—effectively, the “ $c + 1$ ”th class in a c -class model—proves to be a dependable signal for *unknown* objects (0.67 AUC), but less so for *known* ones (0.79 AUC). The most reliable method, however, is derived by amalgamating the background and objectness scores (*bg+obj.*) through arithmetic mean, resulting in 0.93 AUC for *known* and 0.7 AUC for *unknown* objects. This scoring strategy is thus adopted for subsequent experiments.

The takeaway from these experiments is that two-stage object detectors like Mask R-CNN are surprisingly adept at extending their detection capabilities to *unknown* object classes. This suggests an intrinsic duality within such models: they possess an innate “any object” detection ability through the RPN, complemented by an object vs. non-object classification framework.

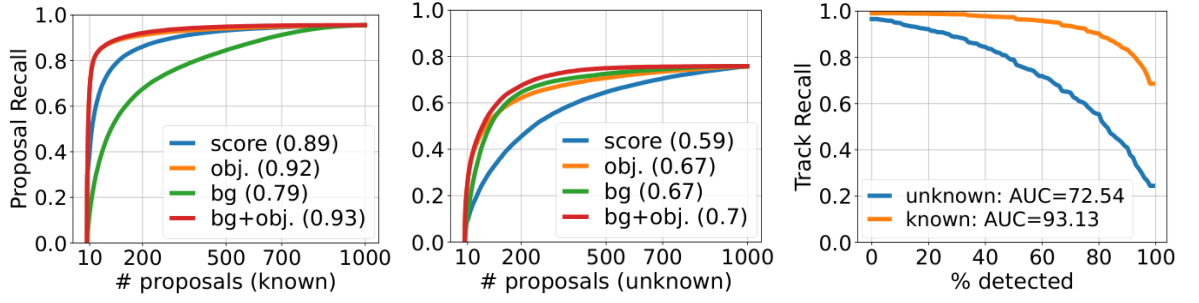


Figure 5.7: Proposal Generation Recall versus Number of Proposals for Different Scoring Methods and Track Recall Analysis The figure provides a detailed comparison of proposal recall against the quantity of proposals generated, evaluating the efficacy of different scoring strategies at an Intersection over Union (IoU) threshold of 0.5. The analysis is segmented into (*left*) recall for *known* objects and (*center*) recall for *unknown* objects. Additionally, (*right*) presents an analysis of track recall across various thresholds of object detection correctness: for instance, a 50% detection rate signifies that at least half of an object’s track is accurately localized within the dataset. This comprehensive view offers insight into how scoring approaches fare in distinguishing objects, especially in differentiating between known and unknown classes.

Track Recall Analysis. In our investigation of the tracking performance, we go beyond mere proposal recall to evaluate the consistency with which *tracks*—the sequences of an object’s positions over time—are correctly estimated. As illustrated in Figure 5.7 (right), we assess the fraction of tracks that meet specific recall criteria across varying percentages of their length. The findings are encouraging, with a remarkable 97% of *unknown* objects being correctly identified at least once along their tracks. Furthermore, a substantial majority, over 80%, are recognized more than half of the time they are present. However, the ability to recall *unknown* objects consistently in every frame they appear is less frequent, with only about 25% achieving this level of recall. This evaluation underscores the strengths and limitations of our tracking approach, highlighting its capability to detect *unknown* objects frequently throughout a video, while also pointing to the need for improvement in continuous track detection.

5.5.2 Evaluating Proposal Similarity for Association (2)

Evaluating Similarity Across Frames. One of the pillars of successful tracking is the ability to determine the similarity between object proposals across different frames, ensuring consistent identity tracking of objects. This task is particularly pivotal for accurate, sustained tracking performance over time. To this end, we address the challenge of short-term association—how effectively can we recognize the same object across a temporal gap? We define this as a *relative classification* task: given an object’s proposal at time t , our goal is to discern this proposal from N competing candidates

in a subsequent frame $t + k$, with k being a short temporal interval, typically about 1 second.

Methodological Analysis. We delve into the specifics of our evaluation by examining several similarity estimation techniques that have been previously introduced in the literature (Bewley et al., 2016; Ošep et al., 2018; Bergmann et al., 2019; Dave et al., 2019; Luiten et al., 2020). Our comparative analysis, presented in Table 5.2, scrutinizes each method’s efficacy under a controlled experimental environment. It is important to note that these methods are solely trained on the *known* class dataset and have had no exposure to *unknown* class data during their training phase. This restriction serves to challenge the generalizability and adaptability of the methods to identify and maintain track of objects they were not explicitly trained to recognize.

Exploring Non-Appearance Based Similarity Measures. In our initial experiments, we concentrated on simplistic similarity measures that disregard the visual content of the objects, purely focusing on the spatial overlap. This subset is denoted as the “Appearance-free” group, where we tested the following metrics:

- **Box Intersection-over-Union (IoU):** This is the most straightforward approach, calculating the IoU based on the bounding boxes of proposals.
- **Mask IoU:** A variant of IoU which, instead of using bounding boxes, utilizes the actual pixel masks of the objects for a more precise overlap measurement.
- **Generalized IoU (GIoU):** Introduced by (Rezatofighi et al., 2019), GIoU is an extension of the standard IoU that also accounts for the cases when one bounding box is completely inside the other.

To counteract the possible propagation of errors due to low-quality matches between frames, we also applied a strategy where association was contingent on surpassing a threshold value. Specifically, we propagated through proposals only if the “box IoU” was greater than 0.75, essentially filtering out matches of subpar quality (“box IoU w/ assoc. thresh”).

Furthering our appearance-free methods, we integrated a Kalman Filter (KF) to project the bounding box’s location into frame $t + k$ before calculating the IoU (“KF, Box IoU”). This technique draws on the parameters detailed in (Bewley et al., 2016) and follows the implementation approaches discussed in (Dave et al., 2019, 2020). The KF is well-regarded for its ability to predict the future state of a moving object based on its current state, despite being purely kinematic and not considering any visual characteristics.

Appearance-free	Method	Inter. Known Unkn.		Method	Inter. Known Unkn.		Method	Inter. FKnown Unkn.						
	Box IoU	✓	86.4	70.7	Regress.	Regression	✓	88.2	65.9	Re-Identification	MaskRCNN <i>euclidean</i>	✓	74.3	63.5
		✗	73.2	72.6		✗	74.7	70.3	✗		75.3	73.3		
	Mask IoU	✓	71.6	39.5	KF, Regression	KF, Regression	✓	87.2	65.5	MaskRCNN <i>cosine</i>	✓	73.0	64.4	
		✗	64.4	45.3					✗		75.0	74.2		
	GIoU	✓	86.4	70.5	Box IoU	✓	87.0	67.6	PReMVOS <i>euclidean</i> *	✗	82.3*	77.1*		
		✗	73.0	70.9		✗	80.1	76.8		PReMVOS <i>cosine</i> *	✗	82.7*	77.5*	
	B. IoU + thresh	✓	81.0	74.7	Mask IoU	✓	73.3	40.8	Mix	Flow-Box IoU +				
KF, Box IoU	✓	84.9	69.1	✗		68.4	47.9	MaskRCNN <i>cosine</i>		✗	86.3	81.9		
				GIoU	✗	80.3	75.9							
				Opt. Fl. + Regr.	✓	88.2	65.9							
					✗	81.4	75.3							

Table 5.2: Association Similarity Ablation Study. We display Top-1 accuracy for proposal association classification at 1FPS, evaluating a range of methodologies discussed in the main text. Methods that yielded the best performance are highlighted: **1st place**, **2nd place**, **3rd place**, **4th place**, **5th place**. The “Inter.” column specifies if “intermediate frames” were utilized in the process. An asterisk (*) denotes an oracle that is not compliant with open-world tracking criteria (i.e., trained on *unknown* classes).

Regression Approaches. For appearance-aware motion prediction, we use the object detector’s regressor trained to predict bounding box transformations over time (Bergmann et al., 2019). This is a *Regression* method that projects the location of the object in frame t to frame $t + k$. We enhance this prediction by applying a Kalman Filter (KF) (Bewley et al., 2016) to the bounding box coordinates, which provides a more robust estimate by considering the object’s previous states. This method is referred to as “KF, Regression”.

Flow-based Methods. We further examine the utility of optical flow in capturing motion between frames. By computing the optical flow field between two frames, we can estimate the motion of each pixel and thus warp object proposals from one frame to the next (Luiten* et al., 2020; Luiten et al., 2020). We experiment with using this warped proposal directly (“Flow, IoU”) and as a refinement step before regression (“Flow + Regression”). These methods leverage temporal consistency by assuming that the motion of pixels is smooth and can be predicted from one frame to the next.

Re-Identification. We further investigate appearance-based re-identification (ReID) for similarity estimation, ensuring that the ReID is trained only on *known* classes. We re-purpose the classification layer embedding (1024D) from our detector for the ReID (“MaskRCNN”). We also evaluate a “non-open-world oracle” ReID which is not limited to training on *known* objects (Luiten et al., 2018) (“PReMVOS”).

Intermediate frames. As TAO is annotated at 1 FPS, we evaluate in two settings: direct (comparing frames 1 second apart directly) and continuous, where the similarity

is propagated through intermediate frames (i.e., we estimate similarity in one frame, select the most similar proposal, and repeat for all intermediate frames; denoted “Inter. frames”).

Discussion. Our evaluation reveals some intriguing insights into the performance of different association strategies. Notably, “box IoU” exhibits robust performance with both *known* (86.4%) and *unknown* (70.7%) objects, showing parity with GIoU and surpassing “mask IoU” which suffers due to its susceptibility to occlusions and the complex movements of articulated objects. When employing a regressor trained exclusively on *known* objects (“Regression”), we observe an improved association for *known* objects (88.2%), but a decline in performance for *unknown* objects (65.9%). Integrating a Kalman Filter with the regressor mirrors the “Regression” results and does not exhibit any accuracy gains over Box IoU by itself. Leveraging intermediate frames seems to benefit the accuracy for *known* objects but detrimentally affects the accuracy for *unknown* objects. This drop in accuracy is attributed to the low recall rates for *unknown* classes by detectors, rendering the dense propagation approach vulnerable to drift during proposal propagation across frames. To mitigate this, we introduce the “B. IoU + thresh” strategy, which selectively bypasses frames with subpar matches, boosting the accuracy for *unknown* objects to 74.7%. Optical flow, on the other hand, enhances results across almost all scenarios, and incorporating appearance-based ReID with features from Mask R-CNN provides a modest boost for both *known* and *unknown* objects. The “oracle” PReMVOS ReID system (Luiten et al., 2018) significantly improves the accuracy for *known* objects, though its impact on *unknown* objects is marginal when compared to flow-based methods. Among the methods evaluated, the combination of optical flow and box IoU emerges as particularly promising. We posit that integrating this method with the Mask R-CNN embeddings could offer further benefits and proceed to assess a straightforward averaging of the similarity scores from these two techniques (“Mix”). This hybrid approach markedly outshines other methods for *unknown* objects. Given that eschewing intermediate frames yields satisfactory results—while also offering a computational speedup of approximately $30\times$ —we opt to exclude intermediate frames from subsequent analyses.

5.5.3 Long-term Tracking (3)

In the next phase of our study, we focus on integrating individual object proposals into coherent long-term tracks. We investigate three distinct approaches: (i) the basic Hungarian matching algorithm, (ii) an enhanced Hungarian matching with a “keep alive” feature designed to preserve track continuity in the face of occlusions or intermittent detection losses (Dave et al., 2019), and (iii) the UnOVOST approach (Luiten et al., 2020), which initially generates short track segments using Hungarian matching and subsequently combines these segments in a secondary offline process.

	Method	Known					Unknown					Unknown-Unknown				
		OWTA	D.Re	A.Acc	A.Re	A.Pr	OWTA	D.Re	A.Acc	A.Re	A.Pr	OWTA	D.Re	A.Acc	A.Re	A.Pr
Val	SORT (Bewley et al., 2016)	46.6	67.4	33.7	39.7	56.4	33.9	43.4	30.3	34.2	57.5	–	–	–	–	–
	Tracktor (Bergmann et al., 2019)	57.9	80.2	42.6	43.6	94.4	22.8	54.0	10.0	10.4	96.6	–	–	–	–	–
	OWTB (Ours)	60.2	77.2	47.4	59.1	57.9	39.2	46.9	34.5	42.6	48.9	–	–	–	–	–
	OWTB (w/o N.O.) [†]	60.8	82.0	45.5	57.3	56.3	42.4	58.9	31.5	39.5	46.8	–	–	–	–	–
	AOA* [†] (Du et al., 2021a)	52.8	72.5	39.1	48.8	53.6	49.7	74.7	33.4	41.1	51.1	–	–	–	–	–
	SORT-TAO* [†] (Dave et al., 2020)	54.2	74.0	40.6	45.0	67.3	39.9	68.8	24.1	28.9	51.6	–	–	–	–	–
Test	SORT (Bewley et al., 2016)	46.6	67.1	33.7	39.5	56.0	32.0	42.2	26.0	30.3	53.7	34.3	44.7	28.2	32.5	56.5
	Tracktor (Bergmann et al., 2019)	57.9	79.7	42.9	43.9	94.5	23.8	53.8	11.0	11.4	96.2	26.3	57.9	12.4	12.8	96.2
	OWTB (Ours)	60.3	76.8	47.8	59.4	58.1	38.5	45.9	33.8	42.4	49.0	41.5	48.9	36.5	45.4	52.3

Table 5.3: Evaluation of the Open-World Tracking Baseline (OWTB) on TAO-OW Validation and Test Sets. We present our assessment in terms of the novel OWTA metric and offer additional insights through Detection Recall (D.Re), Association Accuracy (A.Acc), Association Recall (A.Re), and Association Precision (A.Pr) metrics. On the validation set, we juxtapose our definitive OWTB against previous state-of-the-art (SOTA) trackers on TAO-OW. Regarding the test set, it includes the same *Unknown* classes found in the validation set, and extends with *Unknown-Unknown* classes, which are additional *unknown* classes exclusive to the test set. Note that results marked with * denote systems not trained in an open-world setting (trained on *unknown* classes), and [†] indicates results with overlapping data.

Our findings indicate that the “keep alive” strategy, though it achieves a higher association recall (39.7 OWTA for *unknown*), does so at the expense of association precision when compared to simple Hungarian matching (39.8 OWTA for *unknown*). The “Offline tracklet merging” approach supersedes both aforementioned techniques, yielding the highest performance with an OWTA of 40.2 for *unknown* classes.

5.5.4 Overlap Removal (4)

In scenarios of open-world tracking, reliance on object proposals is crucial. These proposals often overlap, presenting multiple possible interpretations of the evidence at hand. For the OWTB task, there is a necessity for a distinct allocation of each video pixel to an object or the background, which complicates the situation.

We examine two primary strategies for managing overlaps. The first, termed “*non-overlap then track*”, addresses overlaps at the proposal stage before proceeding with tracking. The second, “*track then non-overlap*”, follows the methodology of (Ošep et al., 2018) by tracking initially across overlapping proposals and subsequently applying track scoring and suppression within the video context.

While one might assume the second strategy to be superior due to its consideration of temporal information, making the association task more complex, our observations do not indicate a significant difference in performance between the two. Surprisingly, the more straightforward “*non-overlap then track*” approach yields marginally better

results. This contrasts with findings in (Ošep et al., 2018), where the alternative strategy benefited from depth cues and utilized an approach based on quadratic pseudo-boolean optimization as per (Leibe et al., 2008a)—a method impractical for application at this scale.

5.6 Evaluation

With a comprehensive review of different strategies for open-world tracking, we finalize on a tracking system that capitalizes on the combined strengths of optical flow and re-identification (ReID) similarities to score and align object proposals across video frames. This system coalesces individual tracklets into cohesive tracks by employing a merging process. The culmination of our efforts is christened OWTB (Open-World Tracking Baseline), representing our proposed standard for tracking in open-world settings.

Tab. 5.3 presents the comparative performance results of our Open-World Tracking Baseline (OWTB) on the TAO-OW validation dataset. We benchmark OWTB against established trackers like SORT (Bewley et al., 2016) and Tracktor (Bergmann et al., 2019), with each system utilizing the identical proposal set generated as detailed in Sec. 5.5.1.

The results clearly illustrate that OWTB outshines SORT across several metrics, notably in detection recall (with an improvement of +9.6 for *known* and +3.9 for *unknown* objects), association accuracy (an increase of +13.1 for *known* and +3.6 for *unknown* objects), and the overarching OWTA score (with a substantial leap of +13.2 for *known* and +4.9 for *unknown* objects). This underlines two key benefits of our approach: first, the adeptness of OWTB at tracking a higher quantity of *unknown* objects; second, the capability of our system to produce elongated tracks for objects, which is indicative of superior association recall in complex scenarios. Nevertheless, there is a minor trade-off as the association precision decreases slightly for *unknown* objects, which aligns with expectations given the increased object tracking numbers.

Contrastingly, Tracktor (Bergmann et al., 2019) demonstrates commendable association precision, evidenced by a minimal rate of incorrect object merging. However, it falls short in correctly merging objects, reflected by a lower association recall and, therefore, an inferior association accuracy score when compared to OWTB, particularly with *unknown* objects. Interestingly, Tracktor gains an edge in detection recall over OWTB, attributable to its ability to generate an elevated number of proposals per frame than those provisioned for OWTB.

As a benchmark, we draw comparisons to two leading methods, AOA (Du et al., 2021a) and SORT-TAO (Dave et al., 2020), which set the standard on the closed-world TAO. It is important to note, however, that these comparisons serve merely as references, given that these methods are trained on *unknown* classes, thereby disqualifying them as truly open-world trackers. Additionally, their results are characterized by overlaps, which is not the case with our system.

To assess the influence of enforcing non-overlapping tracks, we also evaluate OWTB’s performance without this constraint. The outcome shows a nominal improvement in scores for both *known* and *unknown* classes. Notably, detection recall sees a marked improvement, albeit at the expense of a slight reduction in both association recall and precision. OWTB exhibits superior performance over the former SOTA closed-world trackers for *known* classes, a testament to its robust design. However, when it comes to *unknown* objects, OWTB is outpaced by the oracle methods, which are specifically trained for the detection and tracking of such objects, hence achieving superior detection recall and association accuracy. This underscores the prevailing challenges in open-world tracking.

TAO-OW Test-set Evaluation. We rigorously evaluate OWTB on the TAO-OW validation set, selecting the best design strategies through extensive tuning. In the true spirit of open-world evaluation, we proceed to assess OWTB on the TAO-OW test set, which encompasses *known*, *unknown*, and the particularly challenging *unknown unknown* classes, the latter being absent from the validation set. As depicted in Tab. 5.3, OWTB maintains consistent performance across both validation and test sets for *known* and *unknown* objects. Crucially, it also shows comparable efficacy on *unknown unknown* classes. We are optimistic that our results will pave the way for future research in the TAO-OW benchmark dedicated to Open-World Tracking.

OWTB vs. Previous Closed-World Trackers. The question arises: Can a tracker optimized for open-world scenarios demonstrate competence in conventional closed-world settings? To explore this, we put OWTB to the test on two established tracking benchmarks, DAVIS unsupervised (Caelles et al., 2019) and KITTI-MOTS (Voigtlaender et al., 2019b). For DAVIS, utilizing our proprietary proposal-generation method, OWTB ranks second with a score of 65.5 in $\mathcal{J}$ & $\mathcal{F}$ metrics, surpassing several contemporary methodologies (Song et al., 2018; Ventura et al., 2019; Wang et al., 2019b; Athar et al., 2020; Gowda et al., 2020; Zhou et al., 2020), with the exception of UnOVOST, which holds a slight edge at 67.9 in $\mathcal{J}$ & $\mathcal{F}$. On KITTI-MOTS, employing publicly available detections, OWTB outperforms renowned TrackR-CNN (Voigtlaender et al., 2019b) (with a HOTA of 56.5/41.9) and the recent PointTrack (Xu et al., 2020) (with a HOTA of 61.9/54.4), specifically for the *car* class, achieving a HOTA of 64.0. It ranks second for the *pedestrian* class with a HOTA of 52.7. This collective outperformance, despite other methods being intricately tuned for specific benchmarks and classes, is a strong indicator of the impressive generalization capabilities inherent to our OWTB system.

5.7 Discussion

Contributions. The objective of this Chapter has been to ignite enthusiasm within the tracking community by shedding light on the expansive possibilities of Open-World

Tracking. We have established the framework for Open-World Tracking as a distinct task, underlined its significance, and introduced a dedicated benchmark alongside a comprehensive evaluation methodology.

Our proposed paradigm for addressing open-world tracking has been meticulously examined through an extensive array of design choices. The culmination of this scrutiny is a tracker that exhibits remarkable proficiency in *both* open-world and closed-world scenarios. We are excited that Open-World tracking is now open for business.

Connection to other Chapters. After creating the HOTA metrics as described in the previous Chapter 4, and using them extensively for the RobMOTS (Luiten, 2021) challenge, one of the clearly needed next things in tracking evaluation was to extend the tracking setting from only operating on a small pre-defined set of object classes to working in an open-world setting. This chapter does exactly that by building upon the HOTA metric from the previous chapter. This chapter also seeks to push the computer vision and tracking community towards a more semantic-agnostic definition of tracking. In Chapter 3 we pursue a different direction toward semantic-agnostic tracking, whereas in this chapter we focus on objects, there we manage to achieve semantic-agnostic tracking by tracking as the level of dense scene elements.

Influence. The work in the this chapter has made a large influence in the computer vision community. The paper this chapter is based on was accepted as an oral presentation at the CVPR 2022 conference. Since then the paper has been cited by over 50 further academic papers which have directly built upon the ideas presented here. Since this paper the general tracking and computer vision community has made a large turn towards open-world and generic tracking and computer vision results, largely due to the influence this work has had as well as many other similar works such as the LVIS dataset.

Conclusion

6.1 Summary and Contributions

In this thesis, I have presented a number of contributions towards the progression of the field of Visual Tracking. Specifically I have focused on using Dynamic 3D reconstruction in order to improve tracking performance, as well as focusing on improving the evaluation metrics and benchmarks that are used in tracking research.

Before this thesis, dynamic 3D representations were barely used for visual tracking, with 2D information being the main information used by algorithms. This thesis pioneered two different approaches to lifting this 2D visual information to 3D and then using this for tracking. Both of the per-object dynamic reconstruction of MOTSFusion, to the persistent volumetric representation of Dynamic 3D Gaussians, have been step changes in how the tracking community thinks about approaching the tracking task and how 3D representations can be used for tracking.

Before this thesis, evaluation for multi-object tracking was severely broken in multiple ways. Firstly, there were no metrics which could evaluate all of the different aspects of tracking performance accurately and simultaneously; and second there was no way to reasonably evaluate the ability of algorithms to track generic objects, rather than just a few classes of objects they were trained on. The works presented in this thesis fixed both of these issues with evaluation by introducing both the HOTA evaluation metrics and the Open-World Tracking Benchmark.

6.2 Perspectives

What does the future hold for Visual Tracking? Personally I believe that tracking will move more toward the “tracking-by-differentiable-rendering” approach championed

by my Dynamic 3D Gaussians method presented in Chapter 3, and away from the deep learning based solutions that most tracking methods follow today. I also see tracking moving toward 3D and toward a more holistic evaluation setting, where every dense element of the scene needs to be tracked over a long-term horizon, rather than just loosely grouping and tracking “objects”. Pushing again on the 3D aspect, for many applications I would argue that it actually makes no sense to produce results in 2D pixel space, as the useful applications require 3D understanding of the motion of the scene. However up until now, doing anything in 3D has been incredibly difficult, but with advances like Dynamic 3D Gaussians, now such results are starting to become possible and will be more and more commonplace and start to show better and better results.

*
* *

Bibliography

- Abou-Chakra, Jad; Dayoub, Feras; & Sünderhauf, Niko (2022). Particlenerf: Particle based encoding for online neural radiance fields in dynamic scenes. *arXiv:2211.04041*. (→ [28](#), [31](#), [41](#), [43](#))
- Alexe, Bogdan; Deselaers, Thomas; & Ferrari, Vittorio (2012). Measuring the objectness of image windows. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, **34**(11), 2189–2202. (→ [107](#))
- Andriluka, Mykhaylo; Roth, Stefan; & Schiele, Bernt (2008). People-tracking-by-detection and people-detection-by-tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ [12](#))
- Andriluka, Mykhaylo; Iqbal, Umar; Insafutdinov, Eldar; Pishchulin, Leonid; Milan, Anton; Gall, Juergen; & Schiele, Bernt (2018). Posetrack: A benchmark for human pose estimation and tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5167–5176. (→ [49](#), [53](#))
- Athar, Ali; Mahadevan, Sabarinath; Osep, Aljosa; Leal-Taixé, Laura; & Leibe, Bastian (2020). Stem-seg: Spatio-temporal embeddings for instance segmentation in videos. In *European Conference on Computer Vision (ECCV)*. (→ [105](#), [122](#))
- Athar, Ali; Luiten, Jonathon; Hermans, Alexander; Ramanan, Deva; & Leibe, Bastian (2022). Hodor: High-level object descriptors for object re-segmentation in video learned from static images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3022–3031. (→ [6](#))
- Athar, Ali; Luiten, Jonathon; Voigtlaender, Paul; Khurana, Tarasha; Dave, Achal; Leibe, Bastian; & Ramanan, Deva (2023a). Burst: A benchmark for unifying object recognition, segmentation and tracking in video. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1674–1683. (→ [6](#))
- Athar, Ali; Hermans, Alexander; Luiten, Jonathon; Ramanan, Deva; & Leibe, Bastian (2023b). Tarvis: A unified approach for target-based video segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18738–18748. (→ [6](#))
- Aygün, Mehmet; Ošep, Aljoša; Weber, Mark; Maximov, Maxim; Stachniss, Cyrill; Behley, Jens; & Leal-Taixé, Laura (2021). 4d panoptic lidar segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ [104](#))

- Babaei, Maryam; Li, Zimu; & Rigoll, Gerhard (2019). A dual cnn-rnn for multiple people tracking. *Neurocomputing*, **368**, 69–83. (→ [89](#), [90](#))
- Baisa, Nathanael L. (2018). Online multi-target visual tracking using a hisp filter. In *Proceedings of the 13th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 5: VISAPP*, pages 429–438. (→ [89](#), [90](#))
- Baisa, N. L. (2019). Online multi-object visual tracking using a gm-phd filter with deep appearance learning. In *2019 22th International Conference on Information Fusion (FUSION)*, pages 1–8. (→ [89](#), [90](#))
- Baisa, Nathanael L. & Wallace, Andrew (2019). Development of a n-type gm-phd filter for multiple target, multiple type visual tracking. *Journal of Visual Communication and Image Representation*, **59**, 257 – 271. (→ [89](#), [90](#))
- Bansal, Aayush & Zollhoefer, Michael (2023). Neural pixel composition for 3d-4d view synthesis from multi-views. In *CVPR*, pages 290–299. (→ [30](#))
- Bansal, Ankan; Sikka, Karan; Sharma, Gaurav; Chellappa, Rama; & Divakaran, Ajay (2018). Zero-shot object detection. In *European Conference on Computer Vision (ECCV)*. (→ [107](#), [108](#))
- Bârsan, Ioan Andrei; Liu, Peidong; Pollefeys, Marc; & Geiger, Andreas (2018). Robust dense mapping for large-scale dynamic environments. In *ICRA*, pages 7510–7517. IEEE. (→ [14](#), [32](#))
- Beauchemin, Steven S. & Barron, John L. (1995). The computation of optical flow. *ACM computing surveys (CSUR)*, **27**(3), 433–466. (→ [31](#))
- Bendale, Abhijit & Boulton, Terrance (2015). Towards open world recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ [106](#))
- Bendale, Abhijit & Boulton, Terrance E (2016). Towards open set deep networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ [106](#))
- Bento, José & Zhu, Jia Jie (2016). A metric for sets of trajectories that is practical and mathematically consistent. *arXiv preprint arXiv:1601.03094*. (→ [19](#), [49](#), [50](#))
- Bergmann, Philipp; Meinhardt, Tim; & Leal-Taixé, Laura (2019). Tracking without bells and whistles. In *IEEE International Conference on Computer Vision (ICCV)*. (→ [89](#), [90](#), [102](#), [105](#), [117](#), [118](#), [120](#), [121](#))
- Bernardin, Keni & Stiefelhagen, Rainer (2008). Evaluating multiple object tracking performance: the clear mot metrics. *EURASIP Journal on Image and Video Processing*, **2008**, 1–10. (→ [19](#), [48](#), [51](#), [58](#), [64](#), [76](#))
- Bernardin, Keni; Elbs, Alexander; & Stiefelhagen, Rainer (2006). Multiple object tracking performance metrics and evaluation in a smart room environment. In *Sixth IEEE International Workshop on Visual Surveillance, in conjunction with ECCV*. (→ [3](#))
- Besl, Paul J & McKay, Neil D (1992). A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*. (→ [13](#))
- Bewley, Alex; Ge, Zongyuan; Ott, Lionel; Ramos, Fabio; & Upcroft, Ben (2016). Simple online and realtime tracking. In *ICIP*. (→ [89](#), [90](#), [117](#), [118](#), [120](#), [121](#))
- Bideau, Pia & Learned-Miller, Erik (2016). A detailed rubric for motion segmentation. *arXiv preprint arXiv:1610.10033*. (→ [104](#))

- Bochinski, Erik; Eiselein, Volker; & Sikora, Thomas (2017). High-speed tracking-by-detection without using image information. In *AVSS*. (→ [89](#), [90](#))
- Brasó, Guillem & Leal-Taixé, Laura (2020). Learning a neural solver for multiple object tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ [89](#), [90](#), [105](#))
- Brendel, William; Amer, Mohamed R.; & Todorovic, Sinisa (2011). Multi object tracking as maximum weight independent set. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ [105](#))
- Breunig, Markus M.; Kriegel, Hans-Peter; Ng, Raymond T.; & Sander, Jörg (2000). Lof: Identifying density-based local outliers. In *ACM SIGMOD International Conference on Management of Data*. (→ [15](#), [16](#))
- Caelles, Sergi; Pont-Tuset, Jordi; Perazzi, Federico; Montes, Alberto; Maninis, Kevis-Kokitsi; & Gool, Luc Van (2019). The 2019 DAVIS challenge on VOS: unsupervised multi-object segmentation. *arXiv arXiv:1905.00737*. (→ [104](#), [122](#))
- Caesar, Holger; Bankiti, Varun; Lang, Alex H; Vora, Sourabh; Liong, Venice Erin; Xu, Qiang; Krishnan, Anush; Pan, Yu; Baldan, Giancarlo; & Beijbom, Oscar (2020). nusenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11621–11631. (→ [49](#), [104](#))
- Cao, Ang & Johnson, Justin (2023). Hexplane: A fast representation for dynamic scenes. In *CVPR*, pages 130–141. (→ [28](#), [30](#))
- Carletta, Jean; Ashby, Simone; Bourban, Sebastien; Flynn, Mike; Guillemot, Mael; Hain, Thomas; Kadlec, Jaroslav; Karaiskos, Vasilis; Kraaij, Wessel; Kronenthal, Melissa; et al. (2005). The ami meeting corpus: A pre-announcement. In *International workshop on machine learning for multimodal interaction*, pages 28–39. Springer. (→ [48](#))
- Chang, Ming-Fang; Lambert, John; Sangkloy, Patsorn; Singh, Jagjeet; Bak, Slawomir; Hartnett, Andrew; Wang, De; Carr, Peter; Lucey, Simon; Ramanan, Deva; & Hays, James (2019). Argoverse: 3d tracking and forecasting with rich maps. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ [49](#), [53](#))
- Chen, Jiahui; Sheng, Hao; Zhang, Yang; & Xiong, Zhang (2017). Enhancing detection model for multiple hypothesis tracking. In *BMTT Workshop*. (→ [89](#), [90](#))
- Chen, L.; Ai, H.; Chen, R.; & Zhuang, Z. (2019). Aggregate tracklet appearance features for multi-object tracking. *IEEE Signal Processing Letters*, **26**(11), 1613–1617. (→ [89](#), [90](#))
- Choi, Wongun (2015). Near-online multi-target tracking with aggregated local flow descriptor. *IEEE International Conference on Computer Vision (ICCV)*. (→ [12](#), [24](#), [105](#))
- Chu, Peng & Ling, Haibin (2019). Famnet: Joint learning of feature, affinity and multi-dimensional assignment for online multiple object tracking. In *ICCV*. (→ [89](#), [90](#))
- Dalal, N. & Triggs, B. (2005). Histograms of oriented gradients for human detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ [3](#))
- Dave, Achal; Tokmakov, Pavel; & Ramanan, Deva (2019). Towards segmenting anything that moves. In *ICCV Workshops*. (→ [104](#), [105](#), [113](#), [114](#), [117](#), [119](#))

- Dave, Achal; Khurana, Tarasha; Tokmakov, Pavel; Schmid, Cordelia; & Ramanan, Deva (2020). Tao: A large-scale benchmark for tracking any object. In *ECCV*. (→ [6](#), [49](#), [50](#), [57](#), [58](#), [75](#), [87](#), [88](#), [102](#), [103](#), [104](#), [107](#), [110](#), [117](#), [120](#), [121](#))
- Dendorfer, P.; Rezatofighi, H.; Milan, A.; Shi, J.; Cremers, D.; Reid, I.; Roth, S.; Schindler, K.; & Leal-Taixé, L. (2019). CVPR19 tracking and detection challenge: How crowded can it get? *arXiv:1906.04567 [cs]*. (→ [49](#))
- Dendorfer, P.; Rezatofighi, H.; Milan, A.; Shi, J.; Cremers, D.; Reid, I.; Roth, S.; Schindler, K.; & Leal-Taixé, L. (2020a). Mot20: A benchmark for multi object tracking in crowded scenes. *arXiv:2003.09003*. (→ [49](#))
- Dendorfer, Patrick; Ošep, Aljoša; Milan, Anton; Schindler, Konrad; Cremers, Daniel; Reid, Ian; & Leal-Taixé, Stefan Roth Laura (2020b). Motchallenge: A benchmark for single-camera multiple target tracking. *International Journal of Computer Vision (IJCV)*. (→ [3](#), [102](#), [103](#), [104](#), [107](#), [113](#))
- Dhamija, Akshay; Gunther, Manuel; Ventura, Jonathan; & Boulton, Terrance (2020). The overlooked elephant of object detection: Open set. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*. (→ [114](#))
- Dhamija, Akshay Raj; Günther, Manuel; & Boulton, Terrance E (2018). Reducing network agnostophobia. In *Advances in Neural Information Processing Systems (NIPS)*. (→ [106](#))
- Doersch, Carl; Gupta, Ankush; Markeeva, Larisa; Contente, Adria Recasens; Smaira, Kucas; Aytar, Yusuf; Carreira, Joao; Zisserman, Andrew; & Yang, Yi (2022). Tap-vid: A benchmark for tracking any point in a video. In *NeurIPS Datasets Track*. (→ [31](#))
- Doersch, Carl; Yang, Yi; Vecerik, Mel; Gokay, Dilara; Gupta, Ankush; Aytar, Yusuf; Carreira, Joao; & Zisserman, Andrew (2023). Tapir: Tracking any point with per-frame initialization and temporal refinement. *arXiv:2306.08637*. (→ [31](#))
- Du, Fei; Xu, Bo; Tang, Jiasheng; Zhang, Yuqi; Wang, Fan; & Li, Hao (2021a). 1st place solution to eccv-tao-2020: Detect and represent any object for tracking. *arXiv preprint arXiv: 2101.08040*. (→ [120](#), [121](#))
- Du, Yilun; Zhang, Yinan; Yu, Hong-Xing; Tenenbaum, Joshua B; & Wu, Jiajun (2021b). Neural radiance flow for 4d view synthesis and video processing. In *ICCV*, pages 14304–14314. IEEE Computer Society. (→ [30](#))
- Edward, K Kao; Matthew, P Daggett; & Michael, B Hurley (2009). An information theoretic approach for tracker performance evaluation. In *2009 IEEE 12th International Conference on Computer Vision*, pages 1523–1529. IEEE. (→ [50](#))
- Eiselein, Volker; Arp, Daniel; Pätzold, Michael; & Sikora, Thomas (2012). Real-time multi-human tracking using a probability hypothesis density filter and multiple detectors. In *2012 IEEE Ninth International Conference on Advanced Video and Signal-Based Surveillance*. (→ [89](#), [90](#))
- Ellis, Anna & Ferryman, James (2010). Pets2010 and pets2009 evaluation of results using individual ground truthed single views. In *2010 7th IEEE International Conference on Advanced Video and Signal Based Surveillance*, pages 135–142. IEEE. (→ [49](#), [53](#))
- Endres, Ian & Hoiem, Derek (2010). Category independent object proposals. In *ECCV*. (→ [107](#))

- Ess, A.; Leibe, B.; Schindler, K.; & Gool, L. van (2008). A mobile vision system for robust multi-person tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ [12](#))
- Everingham, M.; Van Gool, L.; Williams, C.K.I.; Winn, J.; & Zisserman, A. (2010). The pascal visual object classes (VOC) challenge. *International Journal of Computer Vision (IJCV)*, **88**(2), 303–338. (→ [58](#), [65](#), [86](#), [88](#))
- Fang, Jiemin; Yi, Taoran; Wang, Xinggang; Xie, Lingxi; Zhang, Xiaopeng; Liu, Wenyu; Nießner, Matthias; & Tian, Qi (2022). Fast dynamic radiance fields with time-aware neural voxels. In *SIGGRAPH Asia 2022 Conference Papers*. (→ [41](#), [43](#))
- Fridovich-Keil, Sara; Meanti, Giacomo; Warburg, Frederik Rahbæk; Recht, Benjamin; & Kanazawa, Angjoo (2023). K-planes: Explicit radiance fields in space, time, and appearance. In *CVPR*, pages 12479–12488. (→ [28](#), [30](#))
- Fu, Z.; Feng, P.; Angelini, F.; Chambers, J.; & Naqvi, S. (2018). Particle phd filter based multiple human tracking using online group-structured dictionary learning. In *IEEE Access*. (→ [89](#), [90](#))
- Fu, Zeyu; Angelini, Federico; Chambers, Jonathon; & Naqvi, Syed Mohsen (2019). Multi-level cooperative fusion of gm-phd filters for online multiple human tracking. *IEEE Transactions on Multimedia*, **21**(9), 2277–2291. (→ [89](#), [90](#))
- Fujita, Osamu (2013). Metrics based on average distance between sets. *Japan Journal of Industrial and Applied Mathematics*, **30**(1), 1–19. (→ [68](#))
- Gao, Hang; Li, Ruilong; Tulsiani, Shubham; Russell, Bryan; & Kanazawa, Angjoo (2022). Monocular dynamic view synthesis: A reality check. *Advances in Neural Information Processing Systems*, **35**, 33768–33780. (→ [30](#))
- Geiger, Andreas; Lenz, Philip; & Urtasun, Raquel (2012). Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ [3](#), [19](#), [49](#), [53](#), [102](#), [103](#), [104](#), [107](#))
- Geiger, Andreas; Lauer, Martin; Wojek, Christian; Stiller, Christoph; & Urtasun, Raquel (2014). 3d traffic scene understanding from movable platforms. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*. (→ [12](#), [105](#))
- Godard, Clément; Mac Aodha, Oisin; Firman, Michael; & Brostow, Gabriel J. (2019). Digging into self-supervised monocular depth prediction. In *IEEE International Conference on Computer Vision (ICCV)*. (→ [22](#))
- Gowda, Shreyank N; Eustratiadis, Panagiotis; Hospedales, Timothy; & Sevilla-Lara, Laura (2020). Alba: Reinforcement learning for video object segmentation. *arXiv preprint arXiv:2005.13039*. (→ [122](#))
- Gunduz, Gultekin & Acarman, Tankut (2018). A lightweight online multiple object vehicle tracking method. In *IVS*. (→ [24](#))
- Gupta, Agrim; Dollar, Piotr; & Girshick, Ross (2019). LVIS: A dataset for large vocabulary instance segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ [107](#))

- Haritaoglu, Ismail; Harwood, David; & Davis, Larry S. (2000). W4: Real-time surveillance of people and their activities. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, **22**, 809–830. (→ [105](#))
- Harley, Adam W.; Fang, Zhaoyuan; & Fragkiadaki, Katerina (2022). Particle video revisited: Tracking through occlusions using point trajectories. In *ECCV*. (→ [31](#), [41](#), [42](#))
- Hartley, Richard & Zisserman, Andrew (2003). *Multiple view geometry in computer vision*. Cambridge university press. (→ [17](#))
- He, K.; Gkioxari, G.; Dollár, P.; & Girshick, R. (2017). Mask R-CNN. In *IEEE International Conference on Computer Vision (ICCV)*. (→ [105](#), [114](#))
- Held, David; Levinson, Jesse; & Thrun, Sebastian (2013). Precision tracking with sparse 3d and dense color 2d data. In *IEEE International Conference on Robotics and Automation (ICRA)*. (→ [13](#))
- Henschel, Roberto; Leal-Taixé, Laura; Cremers, Daniel; & Rosenhahn, Bodo (2018). Fusion of head and full-body detectors for multi-object tracking. *Trajectory CVPRW*. (→ [89](#), [90](#))
- Henschel, Roberto; Zou, Yunzhe; & Rosenhahn, Bodo (2019). Multiple people tracking using body and joint detections. *CVPRW*. (→ [89](#), [90](#), [94](#))
- Huang, Chang; Wu, Bo; & Nevatia, Ramakant (2008). Robust object tracking by hierarchical association of detection responses. In *European Conference on Computer Vision (ECCV)*. (→ [12](#))
- Hwang, Jaedong; Oh, Seoung Wug; Lee, Joon-Young; & Han, Bohyung (2021). Exemplar-based open-set panoptic segmentation network. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ [106](#))
- I., Mustafa; Rünz, Martin; Georgopoulos, Markos; Khakhulin, Taras; Starck, Jonathan; Agapito, Lourdes; & Nießner, Matthias (2023). Humanrf: High-fidelity neural radiance fields for humans in motion. *ACM Transactions on Graphics (TOG)*, **42**(4), 1–12. (→ [30](#))
- Ilg, E.; Saikia, T.; Keuper, M.; & Brox, T. (2018). Occlusions, motion and depth boundaries with a generic network for disparity, optical flow or scene flow estimation. In *European Conference on Computer Vision (ECCV)*. (→ [13](#), [14](#), [22](#))
- Irani, Michal & Anandan, P (1998). A unified approach to moving object detection in 2d and 3d scenes. *TPAMI*, **20**(6), 577–589. (→ [104](#))
- Jain, Lalit P; Scheirer, Walter J; & Boulton, Terrance E (2014). Multi-class open set recognition using probability of inclusion. In *European Conference on Computer Vision (ECCV)*. (→ [106](#))
- Jain, Ramesh & Nagel, H.-H (1979). On the analysis of accumulative difference pictures from image sequences of real world scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, **1**, 206 – 214. (→ [105](#))
- Joo, Hanbyul; Soo Park, Hyun; & Sheikh, Yaser (2014). Map visibility estimation for large-scale dynamic 3d reconstruction. In *CVPR*, pages 1122–1129. (→ [30](#), [32](#), [41](#))
- Joo, Hanbyul; Liu, Hao; Tan, Lei; Gui, Lin; Nabbe, Bart; Matthews, Iain; Kanade, Takeo; Nobuhara, Shohei; & Sheikh, Yaser (2015). Panoptic studio: A massively multiview system for social motion capture. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3334–3342. (→ [29](#), [39](#))

- Joseph, K J; Khan, Salman; Khan, Fahad Shahbaz; & Balasubramanian, Vineeth N (2021). Towards open world object detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ [106](#))
- Karaev, Nikita; Rocco, Ignacio; Graham, Benjamin; Neverova, Natalia; Vedaldi, Andrea; & Rupprecht, Christian (2023). Cotracker: It is better to track together. *arXiv:2307.07635*. (→ [31](#))
- Karunasekera, H.; Wang, H.; & Zhang, H. (2019). Multiple object tracking with attention to appearance, structure, motion and size. *IEEE Access*, 7, 104423–104434. (→ [89](#), [90](#))
- Kasturi, Rangachar; Goldgof, Dmitry; Soundararajan, Padmanabhan; Manohar, V; Boonstra, M; & Korzhova, V (2006). Performance evaluation protocol for face, person and vehicle detection & tracking in video analysis and content extraction (vace-ii). *Computer Science & Engineering University of South Florida, Tampa*. (→ [48](#))
- Kaucic, R.; Perera, A.; Brooksby, G.; Kaufhold, J.; & Hoogs, A. (2005). A unified framework for tracking through occlusions and across sensor gaps. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ [12](#))
- Kay, Will; Carreira, Joao; Simonyan, Karen; Zhang, Brian; Hillier, Chloe; Vijayanarasimhan, Sudheendra; Viola, Fabio; Green, Tim; Back, Trevor; Natsev, Paul; et al. (2017). The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*. (→ [105](#))
- Keetha, Nikhil; Karhade, Jay; Jatavallabhula, Krishna Murthy; Yang, Gengshan; Scherer, Sebastian; Ramanan, Deva; & Luiten, Jonathon (2023). Splatam: Splat, track & map 3d gaussians for dense rgb-d slam. *arXiv preprint arXiv:2312.02126*. (→ [7](#))
- Kerbl, Bernhard; Kopanas, Georgios; Leimkuehler, Thomas; & Drettakis, George (2023). 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics (TOG)*, 42(4), 1–14. (→ [28](#), [31](#), [34](#), [36](#), [38](#), [39](#), [41](#), [42](#), [43](#))
- Keselman, Leonid & Hebert, Martial (2022). Approximate differentiable rendering with algebraic surfaces. In *ECCV*. (→ [31](#))
- Keuper, M.; Tang, S.; Andres, B.; Brox, T.; & Schiele, B. (2018). Motion segmentation and multiple object tracking by correlation co-clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1. (→ [89](#), [90](#))
- Kim, Chanhon; Li, Fuxin; Ciptadi, Arridhana; & Rehg, James M. (2015). Multiple hypothesis tracking revisited. In *IEEE International Conference on Computer Vision (ICCV)*. (→ [12](#), [89](#), [90](#), [105](#))
- Kim, Chanhon; Li, Fuxin; & Rehg, James M (2018). Multi-object tracking with neural gating using bilinear lstm. In *European Conference on Computer Vision (ECCV)*. (→ [89](#), [90](#))
- Kim, Dahun; Woo, Sanghyun; Lee, Joon-Young; & Kweon, In So (2020). Video panoptic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ [104](#))
- Kirillov, Alexander; He, Kaiming; Girshick, Ross; Rother, Carsten; & Dollár, Piotr (2019). Panoptic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9404–9413. (→ [67](#))
- Kristan, Matej; Matas, Jiří; Leonardis, Aleš; Felsberg, Michael; Pflugfelder, Roman; Kämäräinen, Joni-Kristian; Chang, Hyung Jin; Danelljan, Martin; Cehovin, Luka; Lukežič, Alan; Drbohlav, Ondrej; Käpylä, Jani; Häger, Gustav; Yan, Song; Yang, Jinyu; Zhang, Zhongqun; & Fernández,

- Gustavo (2021). The ninth visual object tracking vot2021 challenge results. In *ICCVW*, pages 2711–2738. (→ [3](#), [31](#))
- Kutschbach, Tino; Bochinski, Erik; Eiselein, Volker; & Sikora, Thomas (2017). Sequential sensor fusion combining probability hypothesis density and kernelized correlation filters for multi-object tracking in video data. In *AVSS*. (→ [89](#), [90](#))
- Layne, Ryan; Hannuna, Sion; Camplani, Massimo; Hall, Jake; Hospedales, Timothy M; Xiang, Tao; Mirmehdi, Majid; & Damen, Dima (2017). A dataset for persistent multi-target multi-camera tracking in rgb-d. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 47–55. (→ [50](#))
- Leal-Taixé, Laura; Pons-Moll, Gerard; & Rosenhahn, Bodo (2011). Everybody needs somebody: Modeling social and grouping behavior on a linear programming multiple people tracker. In *ICCV Workshops*. (→ [105](#))
- Leal-Taixé, Laura; Fenzi, Michele; Kuznetsova, Alina; Rosenhahn, Bodo; & Savarese, Silvio (2014). Learning an image-based motion context for multiple people tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ [105](#))
- Leal-Taixé, Laura; Milan, Anton; Reid, Ian; ; Roth, Stefan; & Schindler, Konrad (2015a). MOTChallenge 2015: Towards a benchmark for multi-target tracking. *arXiv preprint arXiv:1504.01942*. (→ [19](#))
- Leal-Taixé, L.; Milan, A.; Reid, I.; Roth, S.; & Schindler, K. (2015b). MOTChallenge 2015: Towards a benchmark for multi-target tracking. *arXiv:1504.01942 [cs]*. (→ [49](#), [53](#))
- Leal-Taixé, Laura; Canton-Ferrer, Cristian; & Schindler, Konrad (2016). Learning by tracking: Siamese cnn for robust target association. *CVPR Workshops*. (→ [105](#))
- Leal-Taixé, Laura; Milan, Anton; Schindler, Konrad; Cremers, Daniel; Reid, Ian; & Roth, Stefan (2017). Tracking the trackers: an analysis of the state of the art in multiple object tracking. *arXiv preprint arXiv:1704.02781*. (→ [49](#), [51](#), [63](#), [95](#))
- Lee, Byungjae; Erdenee, Enkhbayar; Jin, SongGuo; & Rhee, Phill-Kyu (2016). Multi-class multi-object tracking using changing point detection. In *ECCV Workshops*. (→ [24](#))
- Lee, Sangyun & Kim, Euntai (2018). Multiple object tracking via feature pyramid siamese networks. *IEEE access*, **7**. (→ [89](#), [90](#))
- Lee, Seong-Ho; Kim, Myung-Yun; & Bae, Seung-Hwan (2018). Learning discriminative appearance models for online multi-object tracking with appearance discriminability measures. *IEEE Access*, **6**, 67316–67328. (→ [89](#), [90](#))
- Leibe, Bastian; Schindler, Konrad; Cornelis, Nico; & Gool, Luc Van (2008a). Coupled object detection and tracking from static cameras and moving vehicles. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, **30**(10), 1683–1698. (→ [105](#), [121](#))
- Leibe, Bastian; Leonardis, Aleš; & Schiele, Bernt (2008b). Robust object detection with interleaved categorization and segmentation. *International Journal of Computer Vision (IJCV)*, **77**(1-3), 259–289. (→ [102](#))
- Leichter, Ido & Krupka, Eyal (2013). Monotonicity and error type differentiability in performance measures for target detection and tracking in video. *IEEE transactions on pattern analysis and machine intelligence*, **35**(10), 2553–2560. (→ [49](#), [50](#), [51](#), [53](#), [63](#), [77](#))

- Li, Ruilong; Tanke, Julian; Vo, Minh; Zollhofer, Michael; Gall, Jurgen; Kanazawa, Angjoo; & Lassner, Christoph (2022). Tava: Template-free animatable volumetric actors. In *ECCV*. (→ 30)
- Li, Yuan; Huang, Chang; & Nevatia, Ram (2009). Learning to associate: Hybridboosted multi-target tracker for crowded scene. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2953–2960. IEEE. (→ 50, 51)
- Li, Zhengqi; Niklaus, Simon; Snavely, Noah; & Wang, Oliver (2021). Neural scene flow fields for space-time view synthesis of dynamic scenes. In *CVPR*, pages 6498–6508. (→ 30)
- Lin, Tsung-Yi; Maire, Michael; Belongie, Serge; Hays, James; Perona, Pietro; Ramanan, Deva; Dollár, Piotr; & Zitnick, C Lawrence (2014). Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer. (→ 58, 86, 88, 103)
- Liu, Qiankun; Liu, Bin; Wu, Yue; Li, Weihai; & Yu, Nenghai (2019a). Real-time online multi-object tracking in compressed domain. *IEEE Access*, 7, 76489–76499. (→ 89, 90)
- Liu, Yu-Lun; Gao, Chen; Meuleman, Andreas; Tseng, Hung-Yu; Saraf, Ayush; Kim, Changil; Chuang, Yung-Yu; Kopf, Johannes; & Huang, Jia-Bin (2023). Robust dynamic radiance fields. In *CVPR*, pages 13–23. (→ 30)
- Liu, Ziwei; Miao, Zhongqi; Zhan, Xiaohang; Wang, Jiayun; Gong, Boqing; & Yu, Stella X (2019b). Large-scale long-tailed recognition in an open world. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ 106)
- Long, Chen; Haizhou, Ai; Zijie, Zhuang; & Chong, Shang (2018). Real-time multiple people tracking with deeply learned candidate selection and person re-identification. In *ICME*. (→ 89, 90)
- Lowe, David G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision (IJCV)*, 60, 91–110. (→ 3)
- Luiten, Jonathon (2021). Robmots challenge. https://eval.vision.rwth-aachen.de/rvsu-workshop21/?page_id=110. (→ 6, 123)
- Luiten, Jonathon; Voigtlaender, Paul; & Leibe, Bastian (2018). Premvos: Proposal-generation, refinement and merging for video object segmentation. In *Asian Conference on Computer Vision (ACCV)*. (→ 3, 6, 12, 13, 14, 15, 20, 22, 118, 119)
- Luiten, Jonathon; Torr, Philip; & Leibe, Bastian (2019). Video instance segmentation 2019: A winning approach for combined detection, segmentation, classification and tracking. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0. (→ 87)
- Luiten*, Jonathon; Fischer*, Tobias; & Leibe, Bastian (2020). Track to reconstruct and reconstruct to track. *IEEE Robotics and Automation Letters*, 5(2), 1803–1810. (→ 4, 9, 32, 49, 118)
- Luiten, Jonathon; Zulfikar, Idil Esen; & Leibe, Bastian (2020). UnOVOST: Unsupervised offline video object segmentation and tracking. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*. (→ 6, 104, 105, 113, 117, 118, 119)
- Luiten, Jonathon; Ošep, Aljoša; Dendorfer, Patrick; Torr, Philip; Geiger, Andreas; Leal-Taixé, Laura; & Leibe, Bastian (2021). Hota: A higher order metric for evaluating multi-object tracking. *International Journal of Computer Vision (IJCV)*. (→ 5, 47, 105, 108)

- Luiten*, Jonathon; Liu*, Yang; Zulfikar*, Idil Esen; Dave*, Achal; Ramanan, Deva; Leibe, Bastian; Ošep, Aljoša; & Leal-Taixé, Laura (2022). Opening up open-world tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. (→ [5](#), [6](#), [101](#))
- Luiten, Jonathon; Kopanas, Georgios; Leibe, Bastian; & Ramanan, Deva (2024). Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis. In *3DV*. (→ [5](#), [27](#))
- Luo, Wenhan; Xing, Junliang; Milan, Anton; Zhang, Xiaoqin; Liu, Wei; Zhao, Xiaowei; & Kim, Tae-Kyun (2014). Multiple object tracking: A literature review. *arXiv preprint arXiv:1409.7618*. (→ [49](#))
- Luo, Wenjie; Yang, Bin; & Urtasun, Raquel (2018). Fast and furious: Real time end-to-end 3d detection, tracking and motion forecasting with a single convolutional net. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ [13](#))
- Maksai, Andrii & Fua, Pascal (2019). Eliminating exposure bias and metric mismatch in multiple object tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ [49](#), [50](#), [83](#), [89](#), [90](#), [94](#))
- Maksai, Andrii; Wang, Xinchao; Fleuret, Francois; & Fua, Pascal (2017). Non-markovian globally consistent multi-object tracking. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2544–2554. (→ [19](#), [49](#), [50](#), [83](#))
- Manohar, Vasant; Boonstra, Matthew; Korzhova, Valentina; Soundararajan, Padmanabhan; Goldgof, Dmitry; Kasturi, Rangachar; Prasad, Shubha; Raju, Harish; Bowers, Rachel; & Garofolo, John (2006). Pets vs. vace evaluation programs: A comparative study. In *Proceedings of IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*, pages 1–6. (→ [50](#), [62](#))
- Max, Nelson (1995). Optical models for direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics*, 1(2), 99–108. (→ [35](#), [39](#))
- Milan, Anton; Schindler, Konrad; & Roth, Stefan (2013). Challenges of ground truth evaluation of multi-target tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. (→ [49](#), [51](#), [62](#), [63](#))
- Milan, A.; Roth, S.; & Schindler, K. (2014). Continuous energy minimization for multitarget tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 36(1), 58–72. (→ [12](#), [105](#))
- Milan, Anton; Leal-Taixé, Laura; Schindler, Konrad; & Reid, Ian (2015). Joint tracking and segmentation of multiple targets. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ [104](#))
- Milan, A.; Leal-Taixé, L.; Reid, I.; Roth, S.; & Schindler, K. (2016a). MOT16: A benchmark for multi-object tracking. *arXiv:1603.00831*. (→ [49](#), [78](#), [90](#))
- Milan, Anton; Schindler, Konrad; & Roth, Stefan (2016b). Multi-target tracking by discrete-continuous energy minimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 38(10), 2054–2068. (→ [105](#))
- Mildenhall, Ben; Srinivasan, Pratul P.; Tancik, Matthew; Barron, Jonathan T.; Ramamoorthi, Ravi; & Ng, Ren (2020). Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*. (→ [30](#), [31](#), [35](#))

- Miller, Dimity; Nicholson, Lachlan; Dayoub, Feras; & Sünderhauf, Niko (2018). Dropout sampling for robust object detection in open-set conditions. In *IEEE International Conference on Robotics and Automation (ICRA)*. (→ [106](#))
- Mitzel, Dennis & Leibe, Bastian (2012). Taking mobile multi-object tracking to the next level: People, unknown objects, and carried items. In *ECCV*. (→ [13](#))
- Mitzel, Dennis; Horbert, Esther; Ess, Andreas; & Leibe, Bastian (2010). Multi-person tracking with sparse detection and continuous segmentation. In *European Conference on Computer Vision (ECCV)*. (→ [105](#))
- Moosmann, Frank & Stiller, Christoph (2013). Joint self-localization and tracking of generic objects in 3d range data. In *IEEE International Conference on Robotics and Automation (ICRA)*. (→ [105](#))
- Moosmann, Frank; Pink, Oliver; & Stiller, Christoph (2009). Segmentation of 3d lidar data in non-flat urban environments using a local convexity criterion. In *IEEE Intelligent Vehicles Symposium (IV)*. (→ [105](#))
- Müller, Thomas; Evans, Alex; Schied, Christoph; & Keller, Alexander (2022). Instant neural graphics primitives with a multiresolution hash encoding. *ACM Trans. Graph.*, **41**(4), 102:1–102:15. (→ [31](#), [41](#), [43](#))
- Mur-Artal, Raul & Tardós, Juan D (2017). Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *TRO*. (→ [14](#))
- Naphade, Milind; Anastasiu, David C; Sharma, Anuj; Jagrlamudi, Vamsi; Jeon, Hyeran; Liu, Kaikai; Chang, Ming-Ching; Lyu, Siwei; & Gao, Zeyu (2017). The nvidia ai city challenge. In *2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCOM/IOP/SCI)*, pages 1–6. IEEE. (→ [50](#))
- Newcombe, Richard A; Fox, Dieter; & Seitz, Steven M (2015). Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In *CVPR*, pages 343–352. (→ [32](#))
- Nghiem, Anh-Tuan; Bremond, Francois; Thonnat, Monique; & Valentin, Valery (2007). Etiseo, performance evaluation for video surveillance systems. In *2007 IEEE Conference on Advanced Video and Signal Based Surveillance*, pages 476–481. IEEE. (→ [48](#))
- Osep, Aljosa; Voigtlaender, Paul; Luiten, Jonathon; Breuers, Stefan; & Leibe, Bastian (2018). Towards large-scale video object mining. In *ECCV Workshop on Interactive and Adaptive Learning in an Open World*. (→ [106](#))
- Ošep, Aljoša; Mehner, Wolfgang; Mathias, Markus; & Leibe, Bastian (2017). Combined image- and world-space tracking in traffic scenes. In *IEEE International Conference on Robotics and Automation (ICRA)*. (→ [12](#), [13](#), [21](#), [24](#))
- Ošep, Aljoša; Mehner, Wolfgang; Voigtlaender, Paul; & Leibe, Bastian (2018). Track, then decide: Category-agnostic vision-based multi-object tracking. *IEEE International Conference on Robotics and Automation (ICRA)*. (→ [13](#), [21](#), [104](#), [105](#), [106](#), [113](#), [114](#), [117](#), [120](#), [121](#))
- Ošep, Aljoša; Voigtlaender, Paul; Luiten, Jonathon; Breuers, Stefan; & Leibe, Bastian (2019). Large-scale object mining for object discovery from unlabeled video. *ICRA*. (→ [106](#))

- Ošep, Aljoša; Voigtlaender, Paul; Weber, Mark; Luiten, Jonathon; & Leibe, Bastian (2020). 4d generic video object proposals. In *IEEE International Conference on Robotics and Automation (ICRA)*. (→ [105](#))
- Paragios, Nikos & Deriche, Rachid (2000). Geodesic active contours and level sets for the detection and tracking of moving objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, **22**, 266–280. (→ [105](#))
- Park, Keunhong; Sinha, Utkarsh; Hedman, Peter; Barron, Jonathan T.; Bouaziz, Sofien; Goldman, Dan B; Martin-Brualla, Ricardo; & Seitz, Steven M. (2021a). Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields. *ACM Trans. Graph.*, **40**(6). (→ [30](#))
- Park, Keunhong; Sinha, Utkarsh; Barron, Jonathan T; Bouaziz, Sofien; Goldman, Dan B; Seitz, Steven M; & Martin-Brualla, Ricardo (2021b). Nerfies: Deformable neural radiance fields. In *ICCV*, pages 5865–5874. (→ [30](#))
- Perazzi, F.; Pont-Tuset, J.; McWilliams, B.; Van Gool, L.; Gross, M.; & Sorkine-Hornung, A. (2016). A benchmark dataset and evaluation methodology for video object segmentation. In *CVPR*. (→ [31](#))
- Peri, Neehar; Luiten, Jonathon; Li, Mengtian; Ošep, Aljoša; Leal-Taixé, Laura; & Ramanan, Deva (2022). Forecasting from lidar via future object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17202–17211. (→ [7](#))
- Petrović, Đorđe; Mijailović, Radomir; & Pešić, Dalibor (2020). Traffic accidents with autonomous vehicles: type of collisions, manoeuvres and errors of conventional vehicles' drivers. *Transportation research procedia*, **45**, 161–168. (→ [102](#))
- Pinheiro, P.O.; Collobert, R.; & Dollár, P. (2015). Learning to segment object candidates. In *Advances in Neural Information Processing Systems (NIPS)*. (→ [114](#))
- Pinheiro, P.H.O.; Lin, T.-Y.; Collobert, R.; & Dollár, P. (2016). Learning to refine object segments. In *European Conference on Computer Vision (ECCV)*. (→ [105](#), [114](#))
- Pirsiavash, Hamed; Ramanan, Deva; & C.Fowlkes, Charles (2011). Globally-optimal greedy algorithms for tracking a variable number of objects. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ [105](#))
- Prokudin, Sergey; Ma, Qianli; Raafat, Maxime; Valentin, Julien; & Tang, Siyu (2023). Dynamic point fields. *arXiv:2304.02626*. (→ [32](#))
- Pumarola, Albert; Corona, Enric; Pons-Moll, Gerard; & Moreno-Noguer, Francesc (2021). D-nerf: Neural radiance fields for dynamic scenes. In *CVPR*, pages 10318–10327. (→ [30](#))
- Qi, Jiyang; Gao, Yan; Hu, Yao; Wang, Xinggang; Liu, Xiaoyu; Bai, Xiang; Belongie, Serge; Yuille, Alan; Torr, Philip; & Bai, Song (2021). Occluded video instance segmentation. *arXiv preprint arXiv:2102.01558*. (→ [103](#), [110](#))
- Rahmathullah, Abu Sajana; García-Fernández, Ángel F; & Svensson, Lennart (2016). A metric on the space of finite sets of trajectories for evaluation of multi-target tracking algorithms. *arXiv preprint arXiv:1605.01177*. (→ [50](#))
- Reid, Donald (1979). An algorithm for tracking multiple targets. *IEEE transactions on Automatic Control*, **24**(6), 843–854. (→ [48](#), [105](#))

- Ren, Jimmy; Chen, Xiaohao; Liu, Jianbo; Sun, Wenxiu; Pang, Jiahao; Yan, Qiong; Tai, YuWing; & Xu, Li (2017). Accurate single stage detector using recurrent rolling convolution. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ [14](#), [20](#), [22](#), [23](#), [24](#))
- Ren, Shaoqing; He, Kaiming; Girshick, Ross; & Sun, Jian (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (NIPS)*. (→ [24](#), [106](#), [114](#))
- Rezatofighi, Hamid; Tsoi, Nathan; Gwak, JunYoung; Sadeghian, Amir; Reid, Ian; & Savarese, Silvio (2019). Generalized intersection over union: A metric and a loss for bounding box regression. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ [117](#))
- Ristani, Ergys; Solera, Francesco; Zou, Roger; Cucchiara, Rita; & Tomasi, Carlo (2016). Performance measures and a data set for multi-target, multi-camera tracking. In *European Conference on Computer Vision*, pages 17–35. Springer. (→ [3](#), [48](#), [49](#), [50](#), [53](#), [76](#), [83](#), [109](#))
- Runz, Martin; Buffier, Maud; & Agapito, Lourdes (2018). Maskfusion: Real-time recognition, tracking and reconstruction of multiple moving objects. In *ISMAR*. (→ [14](#))
- Russakovsky, Olga; Deng, Jia; Su, Hao; Krause, Jonathan; Satheesh, Sanjeev; Ma, Sean; Huang, Zhiheng; Karpathy, Andrej; Khosla, Aditya; Bernstein, Michael; Berg, Alexander C.; & Fei-Fei, Li (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*. (→ [50](#), [57](#), [58](#), [76](#), [86](#), [87](#), [88](#))
- Sanchez-Matilla, Ricardo & Cavallaro, Andrea (2019). A predictor of moving objects for first-person vision. In *IEEE International Conference on Image Processing (ICIP)*. (→ [89](#), [90](#))
- Sanchez-Matilla, R.; Poiesi, F.; & Cavallaro, A. (2016). Online multi-target tracking with strong and weak detections. In *Computer Vision – ECCV 2016 Workshops*, pages 84–99. (→ [89](#), [90](#))
- Scheidegger, Samuel; Benjaminsson, Joachim; Rosenberg, Emil; Krishnan, Amrit; & Granström, Karl (2018). Mono-camera 3d multi-object tracking using deep learning detections and PMBM filtering. In *IVS*. (→ [24](#))
- Scheirer, Walter J; Rezende Rocha, Anderson de; Sapkota, Archana; & Boulton, Terrance E (2012). Toward open set recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, **35**(7), 1757–1772. (→ [106](#))
- Scheirer, Walter J; Jain, Lalit P; & Boulton, Terrance E (2014). Probability models for open set recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, **36**(11), 2317–2324. (→ [106](#))
- Sharma, Sarthak; Ansari, Junaid; Murthy, J; & Krishna, K (2018). Beyond pixels: Leveraging geometry and shape cues for online multi-object tracking. In *IEEE International Conference on Robotics and Automation (ICRA)*. (→ [12](#), [13](#), [20](#), [21](#), [23](#), [24](#), [25](#))
- Sheng, Hao; Zhang, Yang; Chen, Jiahui; Xiong, Zhang; & Zhang, Jun (2018). Heterogeneous association graph fusion for target association in multiple object tracking. *IEEE Transactions on Circuits and Systems for Video Technology*. (→ [89](#), [90](#))
- Shi, Jianbo & Malik, Jitendra (1998). Motion segmentation and tracking using normalized cuts. In *ICCV*. (→ [104](#))
- Shitrit, Horesh Ben; Berclaz, Jerome; Fleuret, Francois; & Fua, Pascal (2011). Tracking multiple people under global appearance constraints. In *2011 International Conference on Computer Vision*, pages 137–144. IEEE. (→ [19](#), [49](#), [50](#))

- Simon, Tomas; Joo, Hanbyul; & Sheikh, Yaser (2017). Hand keypoint detection in single images using multiview bootstrapping. *CVPR*. (→ [40](#))
- Singer, R; Sea, R; & Housewright, K (1974). Derivation and evaluation of improved tracking filter for use in dense multitarget environments. *IEEE Transactions on Information Theory*, **20**(4), 423–432. (→ [48](#))
- Slavcheva, Miroslava; Baust, Maximilian; Cremers, Daniel; & Ilic, Slobodan (2017). Killingfusion: Non-rigid 3d reconstruction without correspondences. In *CVPR*, pages 1386–1395. (→ [32](#))
- Smith, Kevin; Gatica-Perez, Daniel; Odobez, Jean-Marc; & Ba, Sileye (2005). Evaluating multi-object tracking. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)-Workshops*, pages 36–36. IEEE. (→ [50](#))
- Smith, P & Buechler, G (1975). A branching algorithm for discriminating and tracking multiple objects. *IEEE Transactions on Automatic Control*, **20**(1), 101–104. (→ [48](#))
- Song, Chonghyuk; Yang, Gengshan; Deng, Kangle; Zhu, Jun-Yan; & Ramanan, Deva (2023). Total-recon: Deformable scene reconstruction for embodied view synthesis. In *ICCV*. (→ [30](#))
- Song, Hongmei; Wang, Wenguan; Zhao, Sanyuan; Shen, Jianbing; & Lam, Kin-Man (2018). Pyramid dilated deeper convlstm for video salient object detection. In *ECCV*. (→ [122](#))
- Song, Y. & Jeon, M. (2016). Online multiple object tracking with the hierarchically adopted gm-phd filter using motion and appearance. *IEEE/IEIE The International Conference on Consumer Electronics (ICCE) Asia*. (→ [89](#), [90](#))
- Song, Y.; Yoon, K.; Yoon, Y.; Yow, K.; & Jeon, M. (2019). Online multi-object tracking with gmphd filter and occlusion group management. *IEEE Access*. (→ [89](#), [90](#))
- Stauffer, Chris & Grimson, W. Eric L. (2000). Learning patterns of activity using real-time tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, **22**, 747–757. (→ [105](#))
- Stein, JJ & Blackman, SS (1975). Generalized correlation of multi-target track data. *IEEE Transactions on Aerospace and Electronic Systems*, **11**(6), 1207–1217. (→ [48](#))
- Stiefelhagen, Rainer; Bernardin, Keni; Bowers, Rachel; Garofolo, John; Mostefa, Djamel; & Soundararajan, Padmanabhan (2006). The clear 2006 evaluation. In *International evaluation workshop on classification of events, activities and relationships*, pages 1–44. Springer. (→ [48](#))
- Sun, Pei; Kretschmar, Henrik; Dotiwalla, Xerxes; Chouard, Aurelien; Patnaik, Vijaysai; Tsui, Paul; Guo, James; Zhou, Yin; Chai, Yuning; Caine, Benjamin; et al. (2020). Scalability in perception for autonomous driving: Waymo open dataset. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ [49](#), [53](#), [83](#), [103](#), [104](#), [110](#))
- Tang, Siyu; Andriluka, Mykhaylo; Andres, Bjoern; & Schiele, Bernt (2017). Multiple people tracking by lifted multicut and person re-identification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ [12](#))
- Teichman, Alex; Levinson, Jesse; & Thrun, Sebastian (2011). Towards 3D object recognition via classification of arbitrary object tracks. In *IEEE International Conference on Robotics and Automation (ICRA)*. (→ [105](#))

- Tian, Wei; Lauer, Martin; & Chen, Long (2019). Online multi-object tracking using joint domain information in traffic scenarios. *IEEE Transactions on Intelligent Transportation Systems*, pages 1–11. (→ [24](#))
- Tian, Yicong; Dehghan, Afshin; & Shah, Mubarak (2018). On detection, data association and segmentation for multi-target tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*. (→ [12](#))
- Turki, Haithem; Zhang, Jason Y; Ferroni, Francesco; & Ramanan, Deva (2023). Suds: Scalable urban dynamic scenes. In *CVPR*, pages 12375–12385. (→ [28](#), [30](#))
- Vedula, Sundar; Baker, Simon; Rander, Peter; Collins, Robert; & Kanade, Takeo (1999). Three-dimensional scene flow. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, volume 2, pages 722–729. IEEE. (→ [31](#))
- Ventura, Carles; Bellver, Miriam; Girbau, Andreu; Salvador, Amaia; Marqués, Ferran; & Giró i Nieto, Xavier (2019). RVOS: end-to-end recurrent network for video object segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ [122](#))
- Voigtlaender, Paul; Luiten, Jonathon; & Leibe, Bastian (2019a). Boltvos: Box-level tracking for video object segmentation. *arXiv preprint arXiv:1904.04552*. (→ [6](#))
- Voigtlaender, Paul; Krause, Michael; Osep, Aljosa; Luiten, Jonathon; Sekar, Berin Balachandrar Gnana; Geiger, Andreas; & Leibe, Bastian (2019b). Mots: Multi-object tracking and segmentation. In *CVPR*, pages 7942–7951. (→ [4](#), [6](#), [12](#), [13](#), [14](#), [19](#), [20](#), [21](#), [23](#), [24](#), [25](#), [31](#), [49](#), [53](#), [71](#), [83](#), [102](#), [104](#), [110](#), [122](#))
- Voigtlaender, Paul; Luiten, Jonathon; Torr, Philip HS; & Leibe, Bastian (2020). Siam r-cnn: Visual tracking by re-detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6578–6588. (→ [6](#))
- Waibel, Alex; Steusloff, Hartwig; Stiefelhagen, Rainer; & Watson, Kym (2009). Computers in the human interaction loop. In *Computers in the Human Interaction Loop*, pages 3–6. Springer. (→ [48](#))
- Wang, Angtian; Wang, Peng; Sun, Jian; Kortylewski, Adam; & Yuille, Alan (2022). Voge: a differentiable volume renderer using gaussian ellipsoids for analysis-by-synthesis. In *ICLR*. (→ [31](#))
- Wang, Dominic Zeng; Posner, Ingmar; & Newman, Paul (2012). What could move? Finding cars, pedestrians and bicyclists in 3D laser data. In *IEEE International Conference on Robotics and Automation (ICRA)*. (→ [105](#))
- Wang, Gaoang; Wang, Yizhou; Zhang, Haotian; Gu, Renshu; & Hwang, Jenq-Neng (2019a). Exploit the connectivity: Multi-object tracking with trackletnet. In *ACM International Conference on Multimedia*, pages 482–490. (→ [49](#), [50](#), [83](#), [89](#), [90](#))
- Wang, Qianqian; Chang, Yen-Yu; Cai, Ruojin; Li, Zhengqi; Hariharan, Bharath; Holynski, Aleksander; & Snavely, Noah (2023). Tracking everything everywhere all at once. *arXiv:2306.05422*. (→ [30](#), [31](#), [41](#))
- Wang, S. & Fowlkes, C. (2016). Learning optimal parameters for multi-target tracking with contextual interactions. *IJCV*. (→ [24](#))

- Wang, Wenguan; Song, Hongmei; Zhao, Shuyang; Shen, Jianbing; Zhao, Sanyuan; Hoi, Steven CH; & Ling, Haibin (2019b). Learning unsupervised video object segmentation through visual attention. In *CVPR*. (→ [122](#))
- Wang, Weiyao; Feiszli, Matt; Wang, Heng; & Tran, Du (2021). Unidentified video objects: A benchmark for dense, open-world segmentation. *arXiv preprint arXiv:2104.04691*. (→ [105](#))
- Wang, Xiaoyu; Yang, Ming; Zhu, Shenghuo; & Lin, Yuanqing (2013). Regionlets for generic object detection. In *IEEE International Conference on Computer Vision (ICCV)*. (→ [24](#))
- Wang, Xueyang; Zhang, Xiya; Zhu, Yinheng; Guo, Yuchen; Yuan, Xiaoyun; Xiang, Liuyu; Wang, Zerun; Ding, Guiguang; Brady, David; Dai, Qionghai; et al. (2020). Panda: A gigapixel-level human-centric video dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3268–3278. (→ [49](#), [83](#))
- Wang, Zhou; Bovik, Alan C; Sheikh, Hamid R; & Simoncelli, Eero P (2004). Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, **13**(4), 600–612. (→ [41](#))
- Weber, Mark; Luiten, Jonathon; & Leibe, Bastian (2020). Single-shot panoptic segmentation. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8476–8483. IEEE. (→ [7](#))
- Weber, Mark; Xie, Jun; Collins, Maxwell; Zhu, Yukun; Voigtlaender, Paul; Adam, Hartwig; Green, Bradley; Geiger, Andreas; Leibe, Bastian; Cremers, Daniel; Ošep, Aljoša; Leal-Taixé, Laura; & Liang-Chieh, Chen (2021). STEP: segmenting and tracking every pixel. In *NeurIPS Track on Datasets and Benchmarks*. (→ [104](#))
- Wedel, Andreas; Brox, Thomas; Vaudrey, Tobij; Rabe, Clemens; Franke, Uwe; & Cremers, Daniel (2011). Stereoscopic scene flow computation for 3d motion understanding. *International Journal of Computer Vision (IJCV)*. (→ [13](#))
- Wen, Longyin; Du, Dawei; Cai, Zhaowei; Lei, Zhen; Chang, Ming-Ching; Qi, Honggang; Lim, Jongwoo; Yang, Ming-Hsuan; & Lyu, Siwei (2020). Ua-detrac: A new benchmark and protocol for multi-object detection and tracking. *Computer Vision and Image Understanding*, **193**, 102907. (→ [49](#), [50](#), [104](#))
- Weng, Chung-Yi; Curless, Brian; Srinivasan, Pratul P; Barron, Jonathan T; & Kemelmacher-Shlizerman, Ira (2022). Humannerf: Free-viewpoint rendering of moving people from monocular video. In *CVPR*, pages 16210–16220. (→ [30](#))
- Weng, Xinshuo; Wang, Jianren; Held, David; & Kitani, Kris (2020). 3d multi-object tracking: A baseline and new evaluation metrics. In *IROS*. (→ [50](#), [76](#))
- Wong, Kelvin; Wang, Shenlong; Ren, Mengye; Liang, Ming; & Urtasun, Raquel (2020). Identifying unknown instances for autonomous driving. In *Conference on Robot Learning*. (→ [107](#), [108](#))
- Wren, Christopher Richard; Azarbayejani, Ali; Darrell, Trevor; & Pentland, Alex (1997). Pfunder: Real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, **19**, 780–785. (→ [105](#))
- Wu, Bo & Nevatia, Ram (2006). Tracking of multiple, partially occluded humans based on static body part detection. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 1, pages 951–958. IEEE. (→ [50](#), [51](#))

- Wu, Chih-Wei; Zhong, Meng-Ting; Tsao, Yu; Yang, Shao-Wen; Chen, Yen-Kuang; & Chien, Shao-Yi (2017). Track-clustering error evaluation for track-based multi-camera tracking system employing human re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1–9. (→ [50](#))
- Wu, Yi; Lim, Jongwoo; & Yang, Ming-Hsuan (2013). Online object tracking: A benchmark. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2411–2418. (→ [3](#))
- Xian, Wenqi; Huang, Jia-Bin; Kopf, Johannes; & Kim, Changil (2021). Space-time neural irradiance fields for free-viewpoint video. *CVPR*. (→ [30](#))
- Xiang, Yu; Alahi, Alexandre; & Savarese, Silvio (2015). Learning to track: Online multiobject tracking by decision making. In *IEEE International Conference on Computer Vision (ICCV)*. (→ [24](#))
- Xie, Christopher; Xiang, Yu; Harchaoui, Zaid; & Fox, Dieter (2019). Object discovery in videos as foreground motion clustering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ [105](#))
- Xu, Jiarui; Cao, Yue; Zhang, Zheng; & Hu, Han (2019). Spatial-temporal relation networks for multi-object tracking. *IEEE International Conference on Computer Vision (ICCV)*. (→ [89](#), [90](#))
- Xu, Ning; Yang, Linjie; Fan, Yuchen; Yang, Jianchao; Yue, Dingcheng; Liang, Yuchen; Price, Brian; Cohen, Scott; & Huang, Thomas (2018). Youtube-vos: Sequence-to-sequence video object segmentation. In *European Conference on Computer Vision (ECCV)*. (→ [103](#))
- Xu, Zhenbo; Zhang, Wei; Tan, Xiao; Yang, Wei; Huang, Huan; Wen, Shilei; Ding, Errui; & Huang, Liusheng (2020). Segment as points for efficient online multi-object tracking and segmentation. In *European Conference on Computer Vision (ECCV)*. (→ [122](#))
- Yang, Gengshan; Vo, Minh; Neverova, Natalia; Ramanan, Deva; Vedaldi, Andrea; & Joo, Hanbyul (2022). Banmo: Building animatable 3d neural models from many casual videos. In *CVPR*, pages 2863–2873. (→ [30](#))
- Yang, Linjie; Fan, Yuchen; & Xu, Ning (2019). Video instance segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5188–5197. (→ [50](#), [53](#), [57](#), [58](#), [75](#), [88](#), [104](#), [105](#), [109](#), [110](#))
- Yoon, Ju Hong; Lee, Chang Ryeol; Yang, Ming Hsuan; & Yoon, Kuk Jin (2016). Online multi-object tracking via structural constraint event aggregation. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ [24](#))
- Yoon, K; Gwak, J; Song, Y; Yoon, Y; & Jeon, M (2020). Oneshotda: Online multi-object tracker with one-shot-learning-based data association. *IEEE Access*, **8**, 38060–38072. (→ [89](#), [90](#))
- Yoon, Youngchul; Boragule, Abhijeet; Song, Youngmin; Yoon, Kwangjin; & Jeon, Moongu (2018). Online multi-object tracking with historical appearance matching and scene adaptive detection filtering. In *IEEE AVSS*. (→ [89](#), [90](#))
- Young, David P & Ferryman, James M (2005). Pets metrics: On-line performance evaluation service. In *2005 IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, pages 317–324. IEEE. (→ [48](#))
- Yu, Fisher; Chen, Haofeng; Wang, Xin; Xian, Wenqi; Chen, Yingying; Liu, Fangchen; Madhavan, Vashisht; & Darrell, Trevor (2020). Bdd100k: A diverse driving dataset for heterogeneous

- multitask learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2636–2645. (→ [49](#), [83](#), [103](#), [110](#))
- Yu, Shou-I; Meng, Deyu; Zuo, Wangmeng; & Hauptmann, Alexander (2016). The solution path algorithm for identity-aware multi-object tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3871–3879. (→ [19](#), [49](#))
- Zhang, Li; Yuan, Li; & Nevatia, Ramakant (2008). Global data association for multi-object tracking using network flows. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (→ [12](#))
- Zhang, Qiang; Baek, Seung-Hwan; Rusinkiewicz, Szymon; & Heide, Felix (2022). Differentiable point-based radiance fields for efficient view synthesis. In *SIGGRAPH Asia 2022 Conference Papers*, pages 1–12. (→ [28](#), [31](#))
- Zhang, Richard; Isola, Phillip; Efros, Alexei A; Shechtman, Eli; & Wang, Oliver (2018). The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*. (→ [41](#))
- Zheng, Yang; Harley, Adam W.; Shen, Bokui; Wetzstein, Gordon; & Guibas, Leonidas J. (2023). Pointodyssey: A large-scale synthetic dataset for long-term point tracking. In *ICCV*. (→ [31](#), [41](#))
- Zhou, Tianfei; Li, Jianwu; Wang, Shunzhou; Tao, Ran; & Shen, Jianbing (2020). Matnet: Motion-attentive transition network for zero-shot video object segmentation. *IEEE Transactions on Image Processing*, **29**, 8326–8338. (→ [122](#))
- Zhu, Ji; Yang, Hua; Liu, Nian; Kim, Minyoung; Zhang, Wenjun; & Yang, Ming-Hsuan (2018). Online multi-object tracking with dual matching attention networks. In *ECCV*. (→ [89](#), [90](#))
- Zhu, Pengfei; Wen, Longyin; Du, Dawei; Bian, Xiao; Hu, Qinghua; & Ling, Haibin (2020). Vision meets drones: Past, present and future. *arXiv preprint arXiv:2001.06303*. (→ [50](#), [57](#), [87](#), [88](#))
- Zwicker, Matthias; Pfister, Hanspeter; Van Baar, Jeroen; & Gross, Markus (2001). Surface splatting. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 371–378. (→ [35](#))