applied sciences

MDPI

Review

A Review of Deep Learning-Based Visual Multi-Object Tracking Algorithms for Autonomous Driving

Shuman Guo [1,*], Shichang Wang [1], Zhenzhong Yang [1], Lijun Wang [1], Huawei Zhang [2], Pengyan Guo [1], Yuguo Gao [1] and Junkai Guo [1]

[1] School of Mechanical Engineering, North China University of Water Resources and Electric Power, Zhengzhou 457003, China
[2] School of Intelligent Transportation, Yunnan Vocational College of Transportation, Kunming 650500, China
* Correspondence: guoshuman@ncwu.edu.cn

Abstract: Multi-target tracking, a high-level vision job in computer vision, is crucial to understanding autonomous driving surroundings. Numerous top-notch multi-object tracking algorithms have evolved in recent years as a result of deep learning's outstanding performance in the field of visual object tracking. There have been a number of evaluations on individual sub-problems, but none that cover the challenges, datasets, and algorithms associated with visual multi-object tracking in autonomous driving scenarios. In this research, we present an exhaustive study of algorithms in the field of visual multi-object tracking over the last ten years, based on a systematic review approach. The algorithm is broken down into three groups based on its structure: methods for tracking by detection (TBD), joint detection and tracking (JDT), and Transformer-based tracking. The research reveals that the TBD algorithm has a straightforward structure, however the correlation between its individual sub-modules is not very strong. To track multiple objects, the JDT technique combines multi-module joint learning with a deep network framework. Transformer-based algorithms have been explored over the past two years, and they have benefits in numerous assessment indicators, as well as tremendous research potential in the area of multi-object tracking. Theoretical support for algorithmic research in adjacent disciplines is provided by this paper. Additionally, the approach we discuss, which uses merely monocular cameras rather than sophisticated sensor fusion, is anticipated to pave the way for the quick creation of safe and affordable autonomous driving systems.

Keywords: autonomous driving; deep learning; visual multi-object tracking; transformer

Citation: Guo, S.; Wang, S.; Yang, Z.; Wang, L.; Zhang, H.; Guo, P.; Gao, Y.; Guo, J. A Review of Deep Learning-Based Visual Multi-Object Tracking Algorithms for Autonomous Driving. *Appl. Sci.* **2022**, *12*, 10741. https://doi.org/10.3390/app122110741

Academic Editor: Byung-Gyu Kim

Received: 26 September 2022
Accepted: 20 October 2022
Published: 23 October 2022

1. Introduction

The primary area of intelligent and networked development in the vehicle and transportation industries is autonomous driving. AVs have the potential to fundamentally alter transportation systems by averting deadly crashes, providing critical mobility to the elderly and disabled, increasing road capacity, saving fuel, and lowering emissions [1,2]. The vehicle perception system's accurate perception of the environment is essential for safe autonomous driving. The perception of autonomous driving settings depends heavily on object tracking, a high-level vision job in the discipline of computer vision. As a result, the development of an object tracking algorithm ensures the development of an automatic driving system that is both safer and more effective.

This section of the review focuses on multi-object tracking in autonomous driving systems. Multi-object tracking is crucial to ensuring the effectiveness and safety of autonomous driving because it is the fundamental component of the technology. Rarely do objects in traffic situations appear alone. Autonomous driving frequently involves recognizing and tracking many things at once, some of which may be moving in relation to the vehicle or to one another. The majority of techniques in the related literature therefore deal with many objects and attempt to address the multi-object tracking issue. In essence,

the MOT algorithm can be summarized as: given the data collected by one or more sensors, how to identify multiple objects in each frame of data and assign an identity to each object, and match those IDs in subsequent data frames [3]. An example of the output of the MOT algorithm is shown in Figure 1 below.

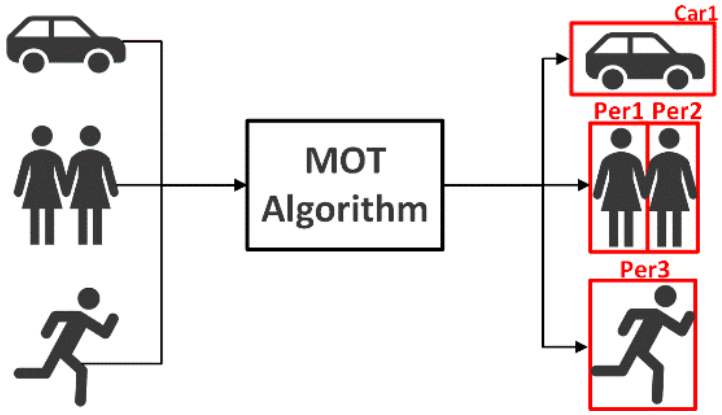

Figure 1. An illustration of the output of an MOT algorithm. Each output bounding box has a number that identifies a specific object in the video.

The research on multi-object tracking for autonomous driving has advanced significantly in recent years, but it is still challenging to use the current multi-object tracking techniques for autonomous driving to their full potential because of issues including the varied shapes of cars and pedestrians in traffic scenes, motion blur, and background interference. There are still several difficulties with the existing visual multi-object tracking technology. Visual multi-object tracking must first address more challenging problems such as: an unpredictable number of objects, frequent object occlusion, challenging object differentiation, etc. In particular, the frequent entry and exit of objects from the field of view is a typical and expected behavior in autonomous driving applications, which results in the uncertainty of the number of objects faced by multi-object tracking and necessitates real-time detection of multi-object tracking algorithms. The method must extract robust object features and keep the object-specific ID after occlusion in complicated situations, since the occlusion of an object by other objects or backgrounds will lead to object ID switches (IDs). The high degree of similarity in object appearance also adds to the difficulty of maintaining the right object ID over the long term. The algorithm must be able to extract the characteristics of comparable items that make them separable. Finally, the challenges that multi-object tracking in autonomous vehicles face can be broken down into two categories: the tracking object factor and the backdrop factor. Shape change, scale change, motion blur, etc. are some of the issues brought on by the object's factors. The impact of backdrop elements is also substantial, particularly the blurring of background interference, occlusion and disappearance of objects, changes in weather, comparable background interference, etc. [4].

This paper introduces algorithms that perform multi-object tracking using the capabilities of deep learning models, concentrating on several approaches for various MOT algorithm frameworks. We concentrate on 2D data extracted from videos shot with a single camera in this evaluation, even though the MOT job may be used with both 2D and 3D data, as well as in single-camera and multi-camera situations. Visual multi-object tracking has made significant strides in recent years due to significant improvements in the detection techniques that use deep learning. Deep learning-based multi-object tracking algorithms have also been used in the area of autonomous driving with considerable success. There are, however, not many more thorough reviews of visual multi-object tracking. Some of the currently used relevant reviews are based on intricate data association algorithms and conventional techniques, which are very unlike the widely used multi-object tracking techniques currently in use [4,5]. A component of the more recent emphasis on detection-based tracking techniques for visual multi-object tracking is based on deep learning [6–9]. As far

as we are aware, there is not a relevant review that compares and contrasts the joint detection and tracking algorithms and the Transformer-based multi-object tracking techniques.

Some reviews and surveys have been published on the subject of MOT. Their main contributions and limitations are the following:

- Luo et al. [5] presented the first in-depth analysis of MOT, particularly pedestrian monitoring, which was groundbreaking. They provided a comprehensive explanation of the MOT problem and outlined the primary methods applied in the crucial MOT system steps. Considering that only a small number of algorithms were using deep learning at the time, they saw it as one of the future research directions. More details were provided regarding complex data association algorithms and conventional methods;
- Ciaparrone et al. [6] offer the first in-depth analysis of how deep learning is used in multi-object tracking. They mainly concentrated on using the MOT algorithm's four phases to apply deep learning techniques. The track management module, nevertheless, which is a crucial component in TBD-based algorithms, was disregarded;
- Sun et al. [7] conducted a thorough analysis of current developments in TBD-based MOT algorithms. They thoroughly described each step of the process and performed a rigorous analysis of the TBD algorithms now in use. The JDT algorithm was regarded as the development trend in the final analysis.

In this paper, based on the discussed limitations, our aim is to provide a survey with the following main contributions:

- In this article, we explore the use of deep learning-based visual multi-object tracking algorithms in autonomous driving. We concentrate on 2D data derived from single-camera movies and discuss current work that has not been previously surveyed or reviewed. In the visual MOT algorithm, we first identified three frameworks: TBD, JDT, and Transformer-based tracking. As a quick reference for future research, we list the various deep learning models and techniques utilized in each framework and illustrate how they differ from one another.
- In order to compare them numerically and uncover the key trends in the top performing methods, we gather experimental results from the most popular MOT datasets.
- As a final point, we discuss the possible future directions of research.

Compared to other studies, this paper offers a thorough overview of the problems, datasets, and algorithms involved in visual multi-object tracking in autonomous driving scenarios. It also offers a theoretical framework for algorithm research by experts in related topics. Additionally, the method we discuss is intended to open a door for the quick development of affordable and safe autonomous driving, because it simply uses monocular cameras as opposed to sophisticated sensor fusion.

2. Methodology

2.1. Review Planning

The study's methodology is based on a systematic review strategy, which is a way to examine and assess the prior literature in relation to the features of the current research. The preparation for the review, the review, and the production of the summary review report are its three main phases. The researchers included 98 studies from the field of visual multi-object tracking that used deep learning in their review of video multi-object tracking under autonomous driving. To finish the article, we first concentrated on the overall autonomous driving field before focusing our search only on the use of deep learning for visual multi-object tracking. The remaining papers were eliminated, and only English full-text publications published in significant journals and conferences between 2012 and 2022 were chosen. To lessen the likelihood of researcher bias, a review strategy must be chosen before a systematic review [10]. We came up with our own overview strategy. First, depending on the goal of the review, we determined the study questions and scope (Section 2.2). In order to identify the final search character and search range, we then executed our search strategy in the database using research questions (Section 2.3). After that,

criteria for research selection were established to decide which publications were included or eliminated from the investigation (Section 2.4). After that, we extracted and categorized the data from the pertinent papers. The multi-object tracking algorithms are divided into TBD, JDT, and Transformer-based tracking methods, based on the outcomes of data categorization and the various network structures of the tracking algorithms (Section 2.5). Finally, in Sections 4–6, we compare and contrast the visual multi-object tracking algorithms of the three frameworks, provide our summaries, and offer our recommendations for the way forward (Section 8).

2.2. Research Objective

In disciplines connected to artificial intelligence, such as visual multi-object tracking, deep learning, and autonomous driving, significant progress has been made in recent years. It becomes more challenging to keep up with developments or enter the area as a newcomer, as is the case with any rapidly expanding field. A thorough analysis of the difficulties, data sources, and algorithms for visual multi-object tracking in autonomous driving scenarios has not yet been published, despite multiple reviews on feature subproblems. By reviewing the most cutting-edge data sets and algorithms, this review aims to close the knowledge gap between learners and the field of automatic driving visual multi-object tracking. Additionally, the approaches included in this evaluation only require monocular cameras as opposed to complicated sensor fusion, which is likely to pave the way for the quick development of more affordable and secure ways to deploy automatic driving systems.

2.3. Search Strategy

We primarily perform our thesis search from two ranges. On the one hand, we gather our research papers from a number of renowned databases. On the other hand, data sets relating to multi-object tracking are searched for our research paper. The sources for our literature search are shown in Table 1 below.

Table 1. Source of paper search.

Source	Inclusion	URL
Thesis Database	Web of Science	https://www.webofscience.com/wos/
	MDPI	https://www.mdpi.com/
	IEEE Xplorer	https://ieeexplore.ieee.org/Xplore/home.jsp
	Springer	https://www.springer.com/gp
	Wiley Library	https://onlinelibrary.wiley.com
	Google Scholar	https://scholar.google.cz/schhp?hl=cs
MOT dataset	MOT16	https://motchallenge.net/results/MOT16/
	MOT17	https://motchallenge.net/results/MOT17/
	MOT20	https://motchallenge.net/data/MOT20/
	KITTI-Tracking	https://www.cvlibs.net/datasets/kitti/eval_tracking.php
	Waymo-Tracking	https://waymo.com/open/
	NuScenes	https://www.nuscenes.org/tracking

We use a Boolean search approach with several AND, OR in the pre-search choices of each data source for the search strategy in each paper database. We used the terms "autonomous driving," "deep learning," "computer vision," and "multi-object tracking" as the major search terms to find papers. According to the goal of this review, we added more keywords to the search to help us find the research paper we were looking for. Based on Boolean operations, the following queries have been created:

- ((Deep Learning) AND (Multi Object Tracking) AND (Computer Vision))
- ((Autonomous Driving) OR (Autonomous Vehicle) OR (Intelligent Vehicle) OR (Self-Driving) AND (Deep Learning) AND (Multi Object Tracking) AND (Computer Vision))
- ((Autonomous Driving) OR (Autonomous Vehicle) OR (Intelligent Vehicle) OR (Self-Driving) AND (Deep Learning) AND (Multi Object Tracking) AND (Computer Vision) AND (Challenge) AND (Development Trend))
- ((Autonomous Driving) OR (Autonomous Vehicle) OR (Intelligent Vehicle) OR (Self-Driving) AND (Deep Learning) AND (Multi Object Tracking) AND (Computer Vision) AND (Dataset) AND (Evaluating Indicator))
- ((Autonomous Driving) OR (Autonomous Vehicle) OR (Intelligent Vehicle) OR (Self-Driving) AND (Deep Learning) AND (Multi Object Tracking) AND (Computer Vision) AND (Tracking by Detection))
- ((Autonomous Driving) OR (Autonomous Vehicle) OR (Intelligent Vehicle) OR (Self-Driving) AND (Deep Learning) AND (Multi Object Tracking) AND (Computer Vision) AND (Joint Detection Tracking))
- ((Deep Learning) AND (Multi Object Tracking) AND (Computer Vision) AND (Transformer))
- ((Autonomous Driving) OR (Autonomous Vehicle) OR (Intelligent Vehicle) OR (Self-Driving) AND (Deep Learning) AND (Multi Object Tracking) AND (Computer Vision) AND (Transformer))

The multi-object tracking dataset frequently includes the method's paper link in addition to the evaluation index of the relevant algorithm. We provide more attention to the publications from the multi-objective data set because, in relative terms, they reflect the most cutting-edge technology at the moment. Here is how we plan to look for relevant publications in the dataset:

1. Determine the data set related to multi-object tracking.
2. Go to the site and view the keyword search related papers from the dataset results, skip this step if the site gives a paper link.
3. Inclusion and exclusion of papers based on keyword; we select the field of visual multi-object tracking in traffic scenarios.
4. The studies that were utilized in this study were released between 2012 and 2022. Using the aforementioned database and datasets, we were able to extract publications. Forty-three research papers were found and chosen during the initial search on the dataset site; 78 papers were found during the search in the aforementioned database. Next, a second search was conducted using the chosen papers to locate any missing research papers from the preliminary analysis. We discovered 21 related research publications using the second search. Thus, 132 papers in total were chosen after combining the two search results. Through the use of multiple inclusion and exclusion criteria, 98 papers from the original 132 publications were ultimately included in the systematic review. The next section will discuss the inclusion and exclusion standards for screening papers.

2.4. Study Selection

We search and choose the research papers, apply the inclusion and exclusion criteria, and collect more authentic and connected research papers in accordance with the review's research objectives and the survey's questions. The following criteria will be used to select which research papers are included and which are excluded:

- Inclusion Criteria:
    1. Research papers on multi-object tracking only applicable to deep learning.
    2. Research papers on multi-object tracking only in traffic scenarios.
    3. Research papers on multi-object tracking only using monocular camera sensors.
- Exclusion Criteria:

1. Research papers on multi-object tracking using traditional methods and complex data association methods.
2. Research paper on single object tracking without extending single object tracking to multi-object tracking.
3. Research Papers Using Sensor Fusion, Not Just Computer Vision.

2.5. Data Extraction and Classification

We eventually acquired 98 research publications by applying the inclusion and exclusion criteria outlined in the previous section. The study goals and research questions listed in Section 2.2 are addressed in these research articles. We categorize these papers into six categories based on our analysis of their application fields and algorithm structures: "autonomous driving development status," "multi-object tracking dataset," "multi-object detection algorithm," "TBD framework algorithm," "JDT framework algorithm," and "Transformer-based tracking framework algorithm" (all research papers use deep learning methods and only use monocular camera sensors). Histograms are used in Figure 2 below to illustrate the data extraction and categorization process.

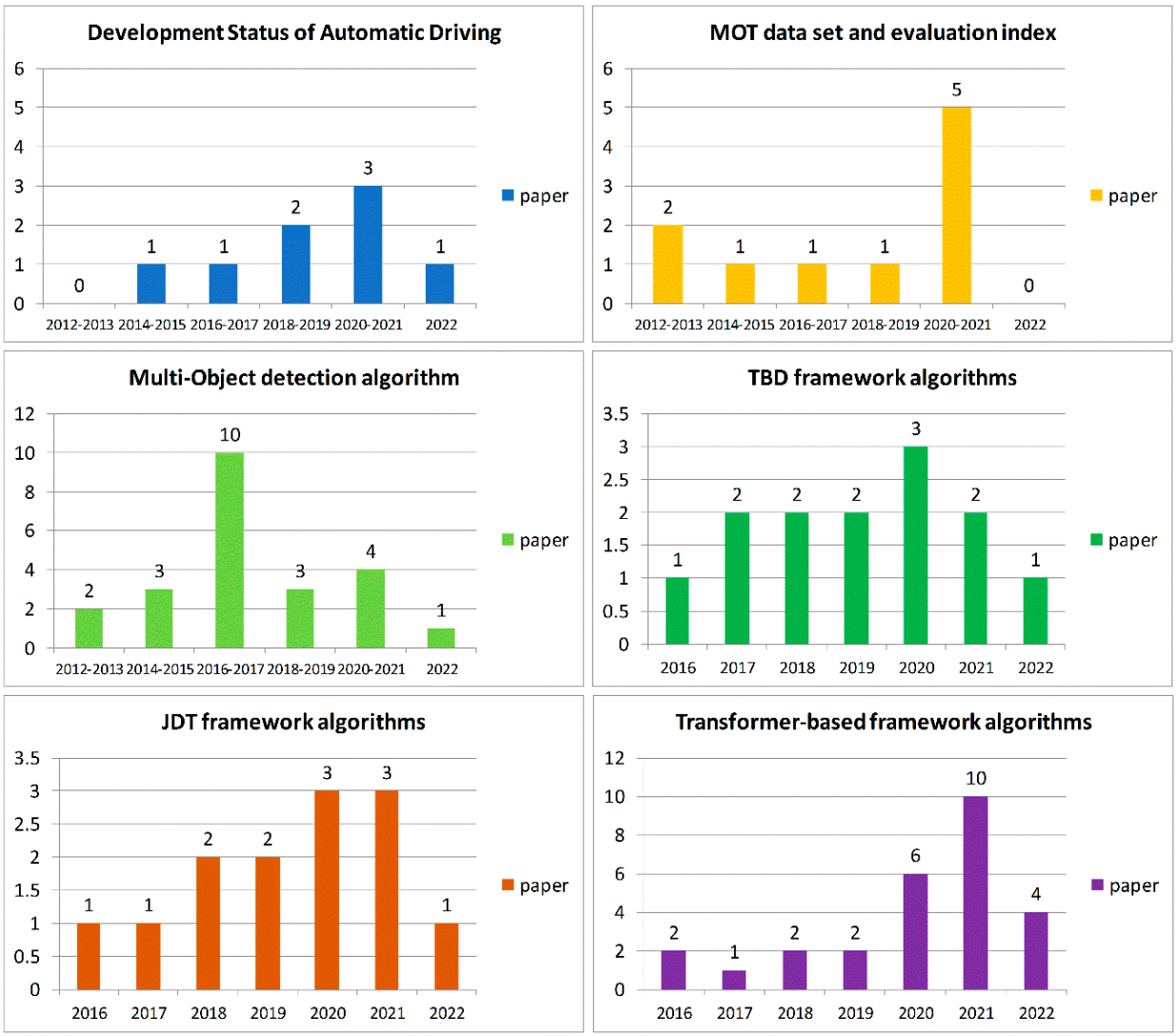

Figure 2. We studied the relevant literature over the past decade, according to the inclusion and exclusion criteria, we finally selected 98 papers as our research object. We divide the paper into six parts and count them in chronological order.

3. Overview of Deep Learning-Based Visual Multi-Object Tracking

Deep learning-based visual multi-object tracking systems have several overview techniques from various angles. The methods for visual multi-object tracking based on deep learning are outlined in this chapter in terms of algorithm classification, related data sets, and algorithm assessment.

3.1. Visual Multi-Object Tracking Algorithm Based on Deep Learning

The tracking algorithm based on detection results has evolved and has quickly taken over as the standard framework for multi-object tracking due to the rapid advancement of object detection algorithm performance [7]. The TBD sub-modules, such as feature extraction, can be included in the object detection network, though, from the standpoint of the deep neural network's structure. Joint detection and tracking, or JDT, using a deep network framework to perform visual multi-object tracking, has emerged as a new development trend based on TBD neutron module fusion [11,12]. The attention mechanism has been incorporated into computer vision systems recently because it has the benefit of efficiently capturing the region of interest in the image, enhancing the performance of the entire network [13–16]. It is used to solve various vision problems, including multi-object tracking. The specific classification structure for the three types of tracking frameworks is shown in Figure 3.

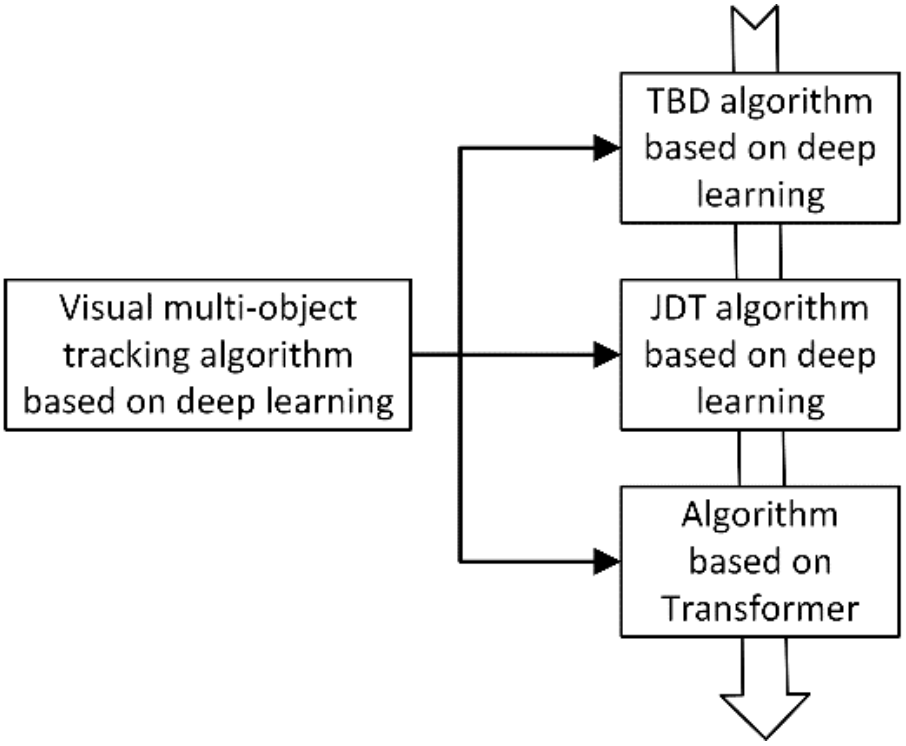

Figure 3. Classification and algorithm of visual multi-object tracking based on deep learning. Overall, the development trend of visual multi-object tracking algorithm is from TBD, to JDT, to Transformer-based tracking algorithm.

At the same time, the characteristics, advantages, and disadvantages of the tracking algorithms of the three types of frameworks in the paper are organized as shown in Table 2 [17–24].

Table 2. Comparison of characteristics of three types of visual multi-object tracking algorithms.

Tracking Algorithm Framework	Principle	Advantage	Disadvantage
TBD	All objects of interest are detected in each frame of the video, and then they are associated with the detected objects in the previous frame to achieve the effect of tracking	Simple structure and strong interpretability	Over-reliance on object detector performance; bloated algorithm design
JDT	End-to-end trainable detection box paradigm to jointly learn detection and appearance features	Multi-module joint learning, weight sharing	Local receptive field, when the object is occluded, the tracking effect is not good
Transformer-based	Transformer encoder-decoder architecture to obtain global and rich contextual interdependencies for tracking	Parallel computing; rich global and contextual information; tracking accuracy and accuracy have been greatly improved, with great potential in the field of computer vision	The parameters are too large and the computational overhead is high; the Transformer-based network has not been fully adapted to the field of computer vision

3.2. MOT Datasets

Deep learning has the benefit over more conventional machine learning techniques in that it can automatically identify data attributes that are pertinent to a specific task. For deep learning-based computer vision algorithms, data sets are crucial. The datasets and traits that are frequently utilized in the field of automatic driving tracking are outlined in the following. Due to their frequent updates and closer resemblance to the actual scene, the MOT datasets [25,26] raise the most concerns in the field of visual multi-object tracking. On the MOT dataset, other cutting-edge tracking methods are also tested.

The MOT16 dataset [25] is used exclusively for tracking pedestrians. There are a total of 14 videos, 7 practice sets, and 7 test sets. These videos were created using a variety of techniques, including fixed and moving cameras, as well as various shooting perspectives. Additionally, the shooting circumstances vary, depending on whether it is day or night and the weather. The MOT16 detector, called DPM, performs better in the area of detecting pedestrians.

The video content of the MOT17 dataset [25] is the same as that of MOT16, but it also provides two detector detection results, namely SDP and Faster R-CNN, which have more accurate ground-truth annotations.

The MOT20 dataset [26] has 8 video sequences, 4 training sets and 4 testing sets, and the pedestrian density is further increased, with an average of 246 pedestrians per frame.

The KITTI dataset [27,28] is currently the largest dataset for evaluating computer vision algorithms in autonomous driving scenarios. These data are used to evaluate 3D object detection and tracking, visual odometry, evaluation of stereo images, and optical flow images.

The NuScenes dataset [29] provides a large dataset of full sensor data for autonomous vehicles, including six cameras, one lidar, five radars, as well as GPS and IMU. Compared with the KITTI dataset, it includes more than seven times more object annotations. For each scene, its key frames are selected for annotation, and the annotation rate is 2 Hz. However, it is worth noting that since 23 types of objects are marked, the class- The imbalance problem will be more serious.

The Waymo dataset [30] is collected with five LiDAR and five high-resolution pinhole cameras. The entire dataset contains 1150 scenes, which are divided into 1000 training sets

and 150 test sets, with a total of about 12 million LiDAR annotation boxes and approx. 12 million image annotation boxes.

The Mapillary Traffic sign dataset [31] is the largest and most diverse traffic sign dataset in the world, which can be used for research on the automatic detection and classification of traffic signs in autonomous driving.

To perform visual multi-object tracking tasks, we gather and introduce the datasets listed in Table 3. Most detection and tracking elements in data collection are related to autos and pedestrians, which helps enhance autonomous driving.

Table 3. Summary of Visual Multi-object Tracking Datasets.

Ref.	Datasets	Year	Feature	DOI/URL
[25,26,32]	MOT15, 16,17,20	2016–2020	Sub datasets containing multiple different camera angles and scenes	https://doi.org/10.48550/arXiv.1504.01942 https://motchallenge.net/
[27,28]	KITTI-Tracking	2012	Provides annotations for cars and pedestrians, scene objects are sparse	https://doi.org/10.1177/0278364913491297 https://www.cvlibs.net/datasets/kitti/eval_tracking.php
[29]	NuScenes	2019	Dense traffic and challenging driving conditions	https://doi.org/10.48550/arXiv.1903.11027 https://www.nuscenes.org/
[30]	Waymo	2020	Diversified driving environment, dense label information	https://doi.org/10.48550/arXiv.1912.04838 https://waymo.com/open/data/motion/tfexample

3.3. MOT Evaluating Indicator

Setting realistic and accurate evaluation metrics is essential for comparing the effectiveness of visual multi-object tracking algorithms in an unbiased and fair manner. The three criteria that make up the multi-object tracking assessment indicators are if the object detection is real-time, whether the predicted position matches the actual position, and whether each object maintains a distinct ID [33]. MOT Challenge offers recognized MOT evaluation metrics.

MOTA (Multi-Object-Tracking Accuracy): the accuracy of multi-object tracking is used to count the accumulation of errors in tracking, including the number of tracking objects and whether they match:

$$\mathrm{MOTP} = \frac{\sum(\mathrm{FN} + \mathrm{FP} + \mathrm{IDSW})}{\sum \mathrm{GT}} \tag{1}$$

where FN (False Negative) is the number of detection frames that do not match the prediction frame; FP (False positive) is the number of prediction frames that do not match the detection frame; IDSW (ID Switch) is the object ID change the number of times; GT (Ground Truth) is the number of tracking objects.

MOTP (Multi-Object-Tracking Precision): the accuracy of multi-object tracking, which is used to evaluate whether the object position is accurately positioned.

$$\mathrm{MOTP} = \frac{\sum \mathrm{Bt}(i)}{\sum \mathrm{C}t} \tag{2}$$

where Ct is the number of matches between the object and the predicted object in the t-th frame; Bt(i) is the distance between the corresponding position of the object in the t-th frame and the predicted position, also known as the matching error.

AMOTA (Average Multiple Object Tracking Accuracy): summarize MOTA over all object confidence thresholds instead of using a single threshold. Similar to mAP for object detection, it is used to evaluate the overall accurate performance of the tracking algorithm under all thresholds to improve algorithm robustness. AMOTA can be calculated by

integrating MOTA under the recall curve, using interpolation to approximate the integral in order to simplify the calculation.

$$\text{AMOTA} = \frac{1}{\text{L}} \sum_{\text{r} \in \{\frac{1}{L}, \frac{2}{L}, \ldots, 1\}} (1 - MOTA_r) \tag{3}$$

where L represents the number of recall values (integration confidence threshold), the higher the L, the more accurate the approximate integral. AMOTA represents the multi-object tracking accuracy at a specific recall value r.

AMOTP (Average Multi-object Tracking Precision): The same calculation method as AMOTA, with recall as the abscissa and *MOTP* as the ordinate, use the interpolation method to obtain AMOTP.

$$\text{AMOTP} = \frac{1}{\text{L}} \sum_{\text{r} \in \{\frac{1}{L}, \frac{2}{L}, \ldots, 1\}} (1 - MOTP_r) \tag{4}$$

IDF1 (ID F1 score): measures the difference between the predicted ID and the correct ID.

MT (Mostly Tracked): the number of objects that are successfully tracked 80% of the time as a percentage of all tracked objects.

ML (Mostly Lost): the percentage of the number of objects that satisfy the tracking success 20% of the time out of all the objects tracked.

FM (Fragmentation): evaluate tracking integrity, defined as FM, counted whenever a trajectory changes its state from tracked to untracked, and the same trajectory is tracked at a later point in time.

HOTA (Higher Order Metric): A higher order metric for evaluating MOT proposed by [34]. Previous metrics overemphasized the importance of detection or association. This evaluation metric explicitly balances the effects of performing accurate detection, association, and localization into a unified metric for comparing trackers. HOTA scores are more consistent with human visual evaluations.

$$HOTA = \int_0^1 HOTA_\alpha d\alpha \tag{5}$$

$$HOTA_\alpha = \sqrt{\frac{\sum_c A(c)}{|TP| + |FN| + |FP|}} \tag{6}$$

$$A(c) = \frac{|TPA(c)|}{|TPA(c)| + |FNA(c)| + |FPA(c)|} \tag{7}$$

where α is the IoU threshold, and c is the number of positive sample trajectories. In the object tracking experiment, there are predicted detection trajectories and ground truth trajectories. The intersection between the two trajectories is called true positive association (*TPA*), and the trajectory outside the intersection in the predicted trajectory is called false positive association (*FPA*). Detections outside the intersection in ground truth trajectories are false negative associations (*FNA*).

4. TBD Algorithms Based Deep Learning

Due to deep learning's potent feature representation capabilities, object detection algorithms have recently performed astronomically better. The result is the creation of a detection-based object tracking technique, which has swiftly emerged as the industry standard for automatic driving multi-object tracking. In order to complete the object tracking task, the MOT algorithm based on TBD first recognizes the object in each frame of the video sequence, extracts the object features, and then associates them in accordance with the feature data. (Figure 4 depicts the MOT flow chart based on the TBD framework).

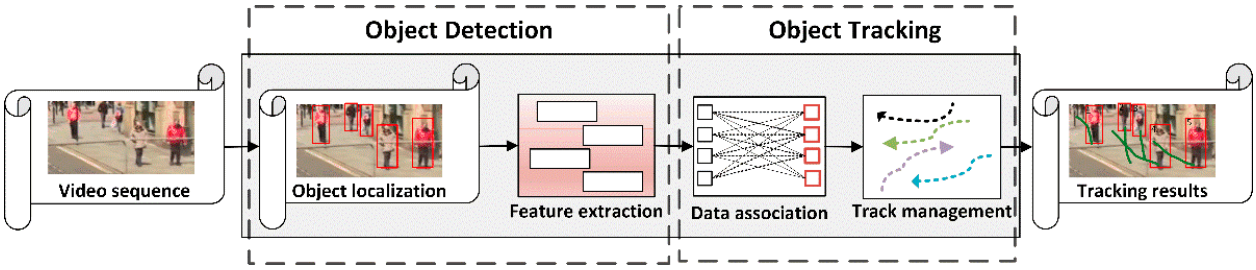

Figure 4. The main procedures of TBD framework, which consists of four core components. The four core components contain object localization, feature extraction, data association, and track management. (Source: This image is from the MOT16 dataset [25]).

4.1. TBD Algorithm Based on Deep Learning Object Detection

Obtaining detection results is the first step in TBD, as already discussed. The results of the object detection have a significant impact on how well the TBD framework tracks objects. There are currently many object detection research findings that are rather advanced [35]. Traditional object detection techniques frequently rely on manually created feature operators to describe images. Examples include SIFT features [36], histogram of gradient directions HOG features [37], etc. However, the results of many related works of literature have demonstrated that compared to traditional algorithms for learning features, Convolutional Neural Networks (CNN) have the strongest feature representation capability and contain the most feature information compared to traditional algorithms learning features.

The mainstream object detectors are R-FCN [38], SSD [39], Faster R-CNN [40], and YOLO series [41–44], which have demonstrated good performance for the detection of objects, such as cars, people, and various obstacles. Neural networks have been used more and more in recent years in the recognition and classification of images. The benefit of using neural networks is their capacity to train on relevant data and learn the depth properties of the tracked objects, allowing for extremely precise detection and categorization. Based on the improved single-frame image detection capabilities, MOT tasks exhibit a trend away from the initial emphasis on computationally demanding data association optimization algorithms, such as Joint Probabilistic Data Association (JPDA) [45] and Multi Hypothesis Tracking (MHT) [46], and toward a TBD framework that depends on detection results.

One of the earliest multi-object tracking algorithms, SORT, uses convolutional neural networks to identify pedestrians. The Faster R-CNN object detection network is used in place of the Aggregate Channel Feature (ACF) detection in the technique, which is based on the conventional Hungarian association approach. The multi-object tracking accuracy was improved on the dataset by an astounding 18.9%, and the algorithm now operates at up to 60 Hz. This algorithm's exceptional performance has garnered a lot of study interest. Aiming at the problem of poor detection quality of Faster R-CNN, Jin et al. [47] extracted multi-scale features and combined three Faster R-CNN models with various backbone structures, which further improved the accuracy and speed of the detection module. They also demonstrated that, after obtaining high-quality detection results, it is possible to simplify the multi-object tracking data association part and obtain comparable multi-object tracking results.

Some researchers have also introduced deep network detectors for visual multi-object tracking, including SSD and YOLO. To complete multi-object tracking tasks, Zhao et al. [48] used a single-stage SSD as a detector and multi-scale augmentation approach training data. The later YOLO version has managed to strike a compromise between object detection accuracy and detection speed and is frequently employed as a detection module for visual multi-object tracking since the YOLO series has a faster detection speed. A multi-object tracking technique based on YOLO was proposed by Li et al. [49]. The video stream was first subjected to multi-object detection using the YOLO algorithm. To exclude extraneous details from the image, depth feature extraction was performed after acquiring the object's

size, location, and other details. The region's noisy data reduces feature extraction's computing difficulty and processing time.

4.2. TBD Algorithm Based on Deep Learning Object Tracking

According to our analysis of the TBD framework method, it consists of two parts: a correlation model and a detection model. This kind of algorithm study is concerned with improving the precision of an object's identity correlation and finding the most effective methods for estimating an object's state using detection information. The procedure of connecting detection results to the trace manager is known as data association. Based on the prerequisite constraint of object detection, object tracking in multi-object tracking systems can be viewed as a problem of finding the best solution. This involves determining the correct motion trajectory of an object using correlation techniques while taking into account factors such as mutual object occlusion, object deformation scale change, etc. Based on the TBD architecture, the following will assess and condense a number of common data association strategies.

4.2.1. SORT-Based Object Tracking

The SORT [50] object tracking algorithm is an online, real-time, multi-object tracking algorithm that incorporates correlation filters into a deep learning algorithm. It predicts the current position using a Kalman filter, correlates the detection frames and objects by correlation, and uses the Intersection over Union (IoU) between each detection and all predicted bounding boxes of an existing object as a metric for the object relationship between the preceding and following frames. Although the object tracking method for SORT is quick, it scarcely addresses object occlusion, leading to a large number of ID switches; accuracy is good in the absence of occlusion but low in the presence of occlusion.

Wojke et al. [51] later proposed the DeepSORT object tracking algorithm in 2017, which extracts the object's apparent features for cascade matching, improves object tracking in the presence of occlusion, and also lessens the issue of object ID switches, to make up for these shortcomings of the SORT algorithm. The main idea behind this approach is to combine frame-by-frame data association, recursive Kalman filtering, and a conventional single-hypothesis tracking method. In order to enhance the ranking algorithm's efficiency and reduce the frequency of ID switches, DeepSORT adds a pre-trained appearance vector based on ResNet [52] network extraction and embeds the cosine distance between features as a cost matrix into the SORT algorithm. Because of this, even if the object is obscured and subsequently reappears, it can still successfully match the ID. Many other researchers adopted comparable CNN networks including GoogleNet [53], ResNet, and Inception-Net [54], as did Mahmoudi et al. [55], Fang et al. [56], Sheng et al. [57], and Chen et al. [48]. By altering the task-related training data, enhancing the associated loss function, learning to identify apparent features that can differentiate between similar objects, and attempting to extract more robust apparent features, the feature extraction backbone network's depth can be increased, but this does not yield much. Lin et al. [58] contend that the TBD algorithm heavily relies on appearance information. As a result, they proposed the hybrid track association (HTA) algorithm, which makes use of the appearance distance in the previous tracking track frame and, by using incremental high number mixture model (IGMM) modeling and incorporating the statistical data derived therefrom, significantly enhances the object recognition. Du et al. [59] optimized DeepSORT in terms of detection, embedding, and association, and proposed the StrongSORT algorithm, which embeds two more lightweight and plug-and-play algorithms, one is an appearance-free link model (AFLink), which associates short trajectories into complete trajectories, the second is to use Gaussian-smoothed interpolation (GSI) to compensate for missed detections and achieve higher accuracy object localization. The StrongSORT algorithm still has several limitations. The main concern is their relatively low running speed compared with joint trackers and several appearance-free separate trackers. Further research on improving computational efficiency is necessary.

4.2.2. LSTM-Based Object Tracking

Multiple studies in the literature have demonstrated that LSTM-based methods have the potential to ensure the correct processing of long-term dependencies while successfully resolving the issue of gradient disappearance and explosion in neural networks through its "gate units." LSTM-related data association techniques have been applied to the field of multi-object tracking [60].

Most LSTM-based techniques for tracking objects combine several classifiers that deal with space and shape with LSTM modules that take temporal consistency into account [61–63]. A long-term tracking solution based on the characteristics collected by the LSTM layer was provided by MILAN et al. [64] who employed an LSTM-based classifier to track objects in a video sequence and realized the re-tracking of items that disappeared and reappeared in the video frequency. While the algorithm performed favorably to other techniques, including the combination of a Kalman filter with the Hungarian algorithm, the results on the MOT15 test set did not quite reach top accuracy; however, the algorithm was able to run much faster than other algorithms (~165 FPS) and did not use any kind of appearance features, leaving room for future improvements. This idea was further explored by [65] who employed a bilinear LSTM network, where one LSTM network tracks motion and the other handles information about object interactions, to identify numerous cues for evaluating long-term relationships. The results of the performance comparison reveal that the approach has stronger robustness and better performance than conventional methods, such as the Hungarian and JPDA algorithms. These two elements are combined to determine the similarity score between frames. Another method of using multiple LSTMs is [66]. Ran et al. proposed a triple-stream network based on pose, which combines three other affinity outputs of three LSTMs to calculate affinity: one for appearance similarity, using CNN features and pose information extracted by AlphaPose [67], one for motion similarity, using posture joint speed, and one for interactive similarity, using an interactive grid. Then, a custom tracking algorithm is used to correlate the detection. When tracking multiple objects, the tracker remembers each object's appearance and motion information. This memory is used to compare trajectory and prediction matching and is updated as a result. In order to overcome the issue of simultaneously taking into account all tracks in the memory update process, Kim et al. [68] added a novel multi-track pooling module to the original model structure, which solved the problem and only added a minor amount of cost.

4.2.3. SORT Tracking vs. LSTM Tracking

This section compares the SORT-based tracking module and the LSTM-based tracking module. In order to maintain fairness, we use the general (public) object detector Faster R-CNN to analyze and compare the results of the MOT17 test set as shown in Table 4, the performance of the deep learning-based visual multi-object tracking algorithm is explored and verified as shown in Table 5.

Table 4. The tracking results of the algorithm based on the TBD framework on the MOT17 test set [25].

Ref.	Methods	Detection	Year.	HOTA	MOTA	IDF1	IDs	FPs
[50]	SORT	Public	2016	34.0	43.1	39.8	4842	143.3
[51]	DeepSORT	Private	2017	54.4	60.3	61.2	2821	20.8
[57]	eHAF17	Public	2018	—	51.8	54.7	1843	0.7
[69]	OCSORT	Public	2021	52.4	58.2	65.1	784	28.6
[59]	StrongSORT	Private	2022	64.4	79.6	79.5	1194	7.1
[65]	MHT_bLSTM	Public	2018	41.0	47.5	51.9	2069	1.9
[68]	BLSTM_MTP	Public	2021	41.3	51.5	51.9	2566	20.1

Table 5. Algorithm analysis and comparison based on TBD framework.

Tracking Module	Ref.	Method	Dataset	Principle	Advantage	Limitation
SORT	[50]	SORT	MOT15 [32]	Input detection object, pass Kalman filter, use IoU as cost matrix to input Hungarian algorithm for object ID matching	The algorithm is simple and runs extremely fast	The association of object IDs is not stable, and the IDs are obvious
	[51]	DeepSORT	MOT16 [25]	On the basis of SORT, the deep apparent feature of the object is added as the association cost	Preliminary realization of the algorithm in the balance of accuracy, speed, IDs	Compared with the latest algorithm, there is a large gap in the accuracy and object IDs indicators
	[59]	StrongSORT	MOT17,20 [25,26]	On the basis of DeepSORT, AFlink module and GSI module are proposed to further improve the tracking results	Further improve the accuracy and precision of the tracking algorithm	Relatively low speed
LSTM	[64]	RNN_LSTM	MOT15 [32]	Online Multi-object Tracking Based on Recurrent Neural Network	LSTM networks are able to learn one-to-one assignments	Poor tracking accuracy and accuracy
	[65]	MHT_bLSTM	MOT16,17 [25]	Evaluating long-term dependencies using a bilinear LSTM network in the MHT framework	Implement LSTM network to model object appearance on long sequences, effectively reducing the frequency of object IDs	Bilinear network brings an increase in the amount of computation, and the computation efficiency is low.
	[68]	BLSTM_MTP	MOT17,20 [25,26]	A multi-track pooling module is added under the bilinear LSTM tracking framework to globally update the appearance and motion information of the object in memory	It performs well in the case of severe occlusion and meets real-time tracking	Tracking accuracy needs to be further improved

The LSTM-based tracker and SORT-based tracker clearly differ from one another, as seen in the above table. While the LSTM tracker performs better when handling occlusion issues, the SORT tracker has an overall edge in tracking speed. Additionally, when the network model's level of detail increases, researchers include various sub-modules with particular roles in the tracker. For instance, StrongSORT augments the DeepSORT tracker with the AFlink and GSI modules, significantly enhancing object tracking's accuracy and precision. However, because of the bloated network structure, the algorithm's processing speed has significantly decreased—to just 7.5 Hz—making it challenging for it to satisfy the demands of real-time applications such as autonomous driving. The detection-based tracking framework is still the most widely used method today, however the goal of research should be to achieve real-time precise tracking of objects without compromising tracking accuracy and precision.

5. JDT Algorithms Based Deep Learning

The study presented above is based on the outcomes of object detection and data association for multi-object tracking. The TBD method's sub-modules (such as feature extraction, etc.) can be integrated into the object detection network, particularly the JDT algorithm, despite the fact that it is still the most widely used algorithm for multi-object tracking.

The joint detection and tracking framework's tracking algorithm has recently undergone new development. The TBD framework becomes less complicated as a result, and multi-object tracking precision is increased. In general, this framework's development will go in one of three directions: first, the detection network will be transformed and integrated into the tracking task so that the designed network model can learn the correla-

tion probability of the object between the sequence frames; second, sub-module fusion in the object detection algorithm or feature fusion will be performed to achieve multi-object tracking tasks; and third, the integration of excellent algorithms in the field of single object tracking will be the focus. Referencing the TBD structure from Figure 4, we choose a few of the sub-modules for fusion in Figure 5 to illustrate how the JDT method works. The JDT method is based on fusing of TBD sub-modules. The purple trapezoidal curve surrounds the object recognition and object tracking module and performs module fusion, which is the first type of JDT algorithm architecture we introduced in Section 5.1. The red rectangular curve surrounds the feature extraction and data association modules and fuses them, which is the second type of JDT algorithm architecture we introduced in Section 5.2. The green elliptic curve surrounds the object positioning, feature extraction, and data association modules, and fuses them. This is the third type of JDT algorithm framework introduced in Section 5.3.

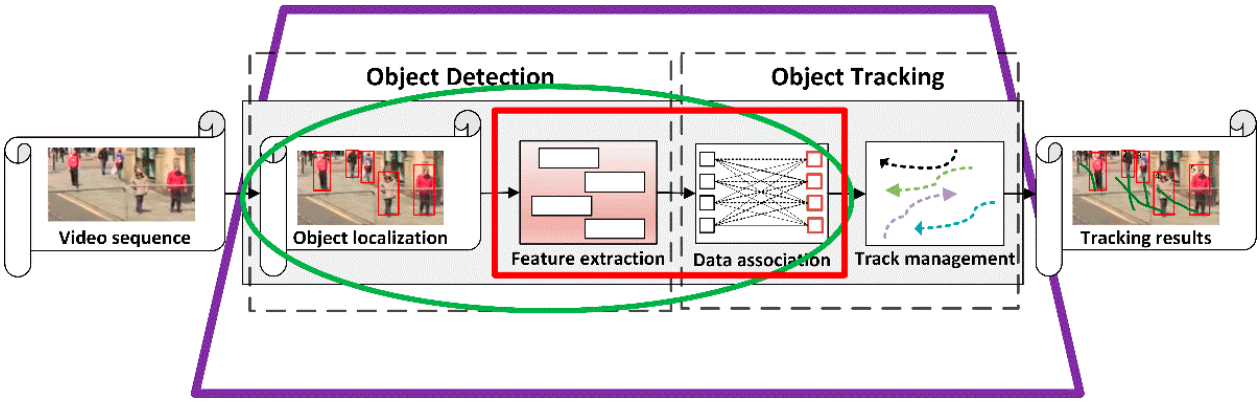

Figure 5. Three Algorithmic Frameworks of JDT. Three different JDT algorithms are created by fusing the sub-modules encompassed by the purple trapezoidal curve, green elliptic curve, and red rectangular curve. (Source: This image is from the MOT16 dataset [25]).

5.1. Fusion Detection and Tracking Module

The front-end object detection network significantly affects how well multi-object tracking functions, as can be shown from the analysis above. In order to further increase algorithm simplicity, weight sharing between the two phases of detection and tracking is realized, and the high-performance object detection network is improved to support multiple objects tracking tasks. Recently, research on object tracking algorithms has turned its attention to fusion of multi-object tracking algorithm modules.

Feichtenhofer et al. [11] initially proposed to include the object detection network in the tracking branch in 2017. They then implemented the main line detection task using an improved R-FCN algorithm, interacted with the multiscale feature maps of the first stage based on the properties of two-stage object detection. This method is based on the traditional twin network framework for single-object tracking, but the original twin network uses 1:1 correlation filtering, whereas the D&T framework uses n:n correlation filtering. The experimentally verified algorithm significantly increases the accuracy and speed of multi-object tracking, but it is still essentially a two-stage tracking algorithm to further integrate the detection and tracking modules. Bergmann et al. [70] proposed a new class of joint detection and tracking framework, Tracktor++, whose core lies in using the tracking frame and observation frame instead of the original RPN module to obtain the true observation frame. Next, data correlation is used to establish the matching of the tracking frame and observation frame. It has been empirically proven that improving the object detection network not only improves the tracking effect but also increases the weight of influence of the fused detection module on the final tracking effect. Inspired by the Tracktor++ framework, Zhang et al. [71] further improved the detection network by adding the predicted optical flow feature module, which turned the tracking frame and

observation frame in Tracktor++ into an optical flow prediction frame and observation frame. After enhancing the motion model, epistatic model, and data association section, Huang et al. [72] also worked to enhance the tracking effect.

When monitoring small objects or a large number of objects, the tracking efficiency is poor, the quantity of calculation is excessive, and the detection speed is too sluggish for automatic driving. The Tracktor++ framework's object recognition techniques all use anchor boxes as their foundation. Given this minimal input to locate objects and predict their associations with the previous frame, Zhou et al.'s multi-object tracking algorithm CenterTrack [73] is applied to the detection of a pair of images and the previous frame. The tracking problem is transformed into the tracking based on the object center point, and the two-dimensional and three-dimensional multi-object tracking of pedestrians and vehicles is realized at the same time.

5.2. Fusion Feature Extraction and Data Association

The deep features retrieved from the object detection network and the deep apparent features used for data association are distinct, as shown by the study of the DBT method. Fusion can boost the neural network's ability to achieve feature fusion and reuse by detecting relevant depth features, REID features, or fusing parent features and motion features.

The above-mentioned Tracktor++ framework is still limited in tracking performance due to the low degree of integration between functional modules. In response to this situation, Peng et al. [74] proposed the CTrack algorithm, which combines the three modules of object detection, feature extraction, and data association. The fusion is integrated into an end-to-end network structure, and CTrack is simple and fast with the help of chain structure and pairwise attention regression technology.

Based on the analysis in the TBD framework, it can be seen that there are differences between the deep features extracted by the detection network and the deep apparent features that the data association relies on. The object detection module fuses appearance features and motion features. Wang et al. [75] proposed a JDE model based on the YOLOv3 detection algorithm. The starting point of the framework is to increase the reusability of features, adding an apparent feature extraction branch to the original classification and regression branch. Although the fusion technique increases multi-object tracking accuracy, it significantly slows down the tracking algorithm.

5.3. Fusion Algorithm of Single Object Tracking

The single object tracking task and the visual multi-object tracking task have a strong relationship. The expected kinematic and visual characteristics of a single object tracking are included in the single object tracking algorithm. Due to the swift advancement of the single object tracking field, the fusion of single object tracking for multi-object tracking has persisted in recent years.

The single object tracking (SOT) method itself has information such as placement and identification thanks to the twin network structure. As a result, several iterations of multi-object tracking algorithms have emerged that leverage the single object tracking technique to replace the motion model and the appearance model. In theory, the multi-object tracking algorithm based on single-object tracking can be compared to the detection-based tracking algorithm, since the problem of lack of observation has some robustness and temporary object positioning information can be gained by regional search.

Zhu et al. [76] proposed the DMAN algorithm to integrate the advantages of single-object tracking and data association methods in a unified framework, and in incorporating SOT into MOT, they introduced cost-sensitive tracking loss for visual tracking to deal with mutual occlusion and interference between objects. Feng et al. [77] proposed the LSST algorithm for the problem of occlusion leading to trajectory features with residuals and even ID switches, and the base tracker is a SiamRPN for fast and accurate detection in the field of single-object tracking, which achieves long-term stable tracking based on the epistatic information extracted by the ReID algorithm. Chu et al. [78] proposed a KCF

algorithm with a relatively complex structure, and designed an Instance-aware SOT tracker by encoding awareness both within and between the object model. Then, the joint model was tested and corrected, the dynamic model was refreshed, and the goal management was among the best in the MOT challenge.

In accordance with the earlier introduction, we discovered that the technology that combines single object tracking and multi-object tracking has both benefits and glaring drawbacks. When the scene composition is modest, the quick and precise feature extraction and positioning capabilities of single object tracking technology are somewhat adaptable for the issues of false detection and missed detection. This is good, but when there are numerous scenario items, an object tracker needs to be added to every item, leading to major efficiency and real-time performance issues that call for a deeper investigation of the method.

5.4. Comparison of JDT Algorithms

We organize the characteristics, advantages, and disadvantages of the typical JDT algorithm in this chapter as shown in Table 6.

Table 6. Algorithm analysis and comparison based on JDT framework.

Ref.	Method	Dataset	Principle	Advantage	Limitation
[70]	Tracktor++	MOT15 [32], MOT16 [25]	Utilize existing deep detection algorithms to integrate tracking functions into detection modules to simplify training on tracking tasks	Using the detection network to achieve the tracking function, the algorithm accuracy and speed have been greatly improved	Too much reliance on detectors
[73]	CenterTrack	MOT16 [25], KITTI [28]	On the basis of CenterNet, add the image information of the previous frame to realize real-time tracking	Run in real time without relying on apparent features and achieves high tracking accuracy	High frequency of IDs for long-term tracking
[75]	JDE	MOT16 [25]	A network for joint learning of detection and embedding of apparent features	Run in real time and is comparable to TBD algorithms in accuracy	The existence of anchor boxes leads to the misalignment of detection and embedded appearance features
[78]	KCF	MOT15 [32], MOT16 [25]	Combining the complementarity of single-object tracking and data association to build end-to-end visual multi-object tracking	When the number of objects is small, it can achieve high tracking accuracy and speed, and has high anti-occlusion performance	Algorithms perform poorly when dealing with objects moving in and out of view quickly and with a large number of objects

When compared to the TBD method, the JDT algorithm performs significantly better since it can carry out end-to-end multi-object tracking and achieve real-time tracking while maintaining a high level of tracking accuracy. The JDT algorithm is built upon the union of the sub-modules of the TBD algorithm. With Tracktor++, the RPN network directly replaces the data association phase in the two-stage detection network, promoting the integration of detection and tracking while increasing the tracking accuracy. The JDT algorithm still has to improve significantly in terms of processing occlusions, speed, and tracking accuracy. However, in general, the JDT algorithm is extended from the TBD method to the multi-object tracking algorithm, and a gradual balance between accuracy and speed is attained. A nice direction is also taken by the multi-object tracking method, which is built on the fusion of single-object tracking. KCF, for instance, excels in the MOT15 and MOT16 data sets, demonstrating to us the potential of extending single-object tracking to multi-object tracking. There is still a problem, though: multi-object tracking based on a single item will add trackers for numerous objects as the number of objects in the scene rises, slowing down the process.

6. MOT Algorithm Based on Transformer

These transformer-based models with excellent representation capabilities have created a breakthrough in natural language processing (NLP). Attention methods were first

presented by [79] for application to machine translation, achieving strong performance in NLP tasks.

The CNN model, which combines a number of linear layers and nonlinear activations with strong feature representation capabilities, is the deep learning-based approach that computer vision experts use the most frequently [35,41,80]. Convolution and pooling layers are introduced by CNN to handle shift-invariant input. Convolution cannot fully utilize contextual information because it lacks a comprehensive grasp of the image itself and is unable to model the relationship between features. Convolution's weights are also fixed and unable to adjust dynamically in response to changes in the input. Recent research has used the transformer in the field of computer vision, inspired by the tremendous success of transformer design in NLP. In contrast to CNN, the transformer's self-attention mechanism is not constrained by local interactions; instead, it can aggregate data from all inputs and achieve fast parallelism. The transformer's perfect memory and global computing power also make it more appropriate for processing lengthy sequences [81]. In order to fully study the characteristics of deep neural networks and enhance the network's accuracy, the depth of the transformer can also be raised.

In the past decade, attention mechanisms have played an increasingly important role in computer vision, and are used to solve a variety of vision problems, including object detection [15,82], semantic segmentation [83,84], 3D vision [85,86], and are starting to achieve success in the field of multi-object tracking.

The multi-object tracking problem includes the transformer architecture, which has shown tremendous success in the CV filed. We present a simplified depiction of the encoder–decoder architecture in Figure 6 to further our understanding of the integration of transformer-based multi-object tracking algorithms for trajectory queries in decoder self-attention blocks. The transformer encoder receives image information from the CNN backbone and adds spatial position encoding to the query and key of each multi-head self-attention layer. The decoder then receives the query (which was initially set to zero), produces the positional encoding (object queries), accesses the encoder memory, and uses multiple multi-head self-attention and decoder attention to produce the final set of predicted class labels and bounding boxes [15].

6.1. MOT Algorithm Based on Transformer Architecture

DETR [15] is the first framework to introduce the transformer successfully into object detection. DETR uses CNN to extract image feature mappings, uses the extracted features as input to the transform encoder, and obtains detection results through the decoder of the transformer. TransTrack [87] introduced the attention mechanism into multi-object tracking for the first time. Firstly, the current frame image was input into the CNN backbone to extract the image feature maps and using the Query–Key mechanism for joint detection and tracking. This algorithm combines the features of two adjacent frames, object queries are used to learn to detect new objects, track queries are responsible for keeping the tracking track, one of the two decoders is responsible for detecting and generating a detection frame, and the other is responsible for generating a tracking frame for object propagation. Finally, the tracking frame and the detection frame are associated on the same frame using the IOU matching strategy to complete the multi-object tracking task. The algorithm first works to solve the MOT task in such a paradigm. It provides a novel perspective for multiple object tracking. Meinhardt et al. [88] proposed the TrackFormer algorithm, which treats multi-object tracking as a set of prediction problems with joint detection and tracking by attention. Similarly, based on the Query–Key mechanism, the architecture consists of a CNN for image feature extraction, a transformer encoder, and a transformer decoder for image coding. The image feature maps extracted using the CNN are encoded by self-attention features in the encoder and the queries are decoded by self-attention and cross-attention in the decoder, resulting in an output embedding with bounding box and class information. The incorporation of the attention mechanism ensures that the model simultaneously considers location, occlusion, and object recognition features and performs well in long

tracking scenarios relative to TransTrack when tested under the dataset. Xu et al. [89] concluded that the bounding box-based tracking methods of both algorithms, TransTrack and TrackFormer, are not suitable for dealing with dense scenarios, and therefore proposed the TransCenter algorithm, which uses a dense multi-scale query to obtain a centroid-based representation of the object heat map and a two-dimensional Gaussian distribution to represent the position and size of the object, and in addition, considering the memory loss, proposed a variable decoder, and finally, combine the central heat map and a bidirectional decoding structure with geometric features and visual features of the decoder through the Hungarian algorithm for object association.

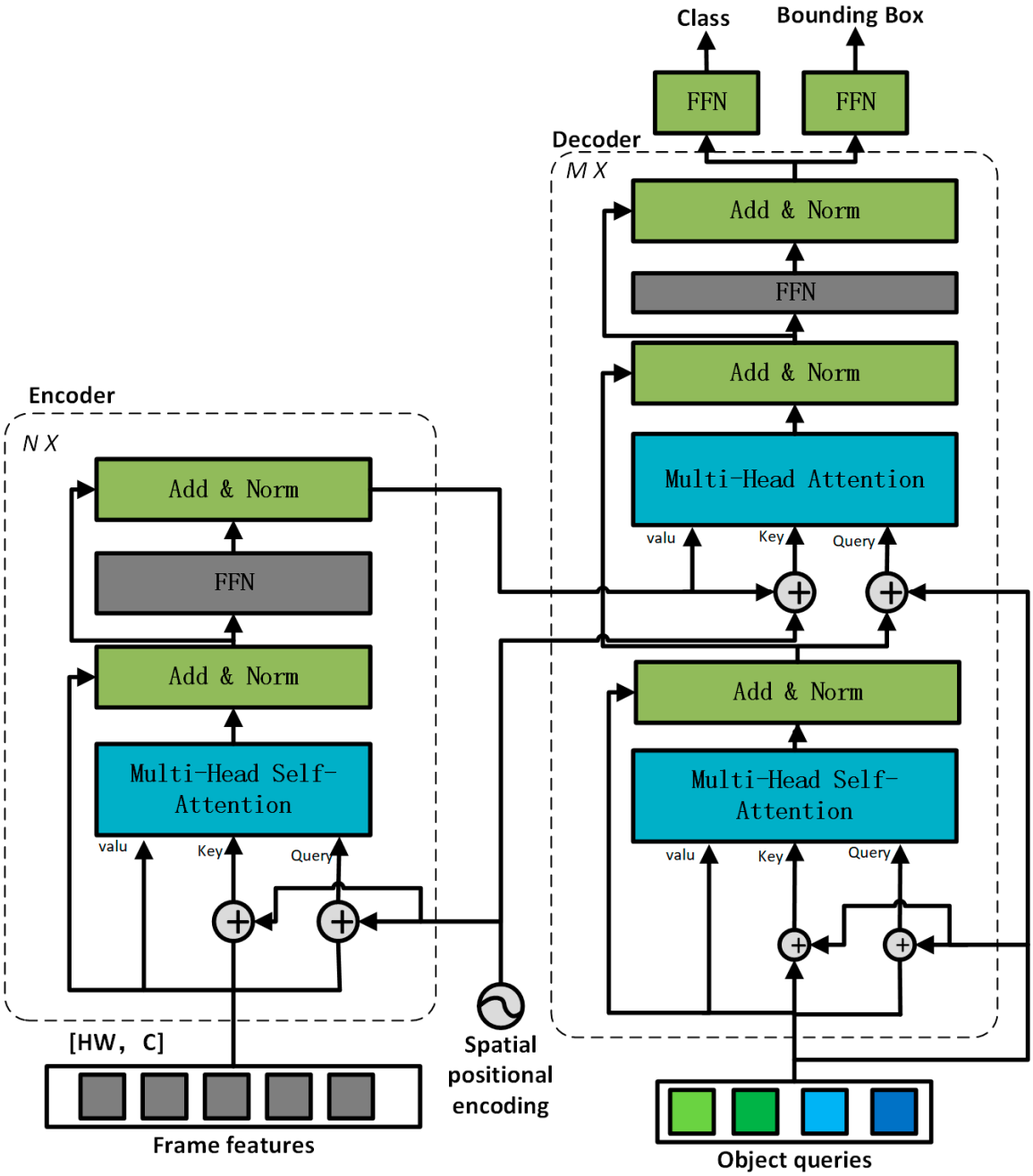

Figure 6. The transformer's encoder–decoder architecture. (Source: Data from DETR [15]).

Chu et al. [90] proposed the TransMOT algorithm, which used the YOLOv5 algorithm as the detector model and the SiamFC network as the visual feature extraction sub-network, and then innovatively proposed a spatial and temporal graph transformer to model multiple objects in space and time, to achieve the purpose of long-term tracking. Specifically, the authors construct a series of sparse weighted graphs based on the spatial relationship of the object. Based on these weighted graphs, a spatial graph attention encoder, a temporal graph attention encoder, and a spatial attention decoder are established for modelling, a cascade association mechanism is also established to deal with low-confidence detections and long-term occlusions, to further optimize the speed and accuracy of TransMOT. Xie et al. [91] proposed a Siamese-like Dual-Branch network based solely on transformers (DualTFR) for visual object tracking in 2021. Dual branches are templates and search images, respectively, divided into non-overlapping facets and extracted a feature vector for each facet based on the results of matching each facet with the others within the attention window. The advantage of the approach is that the features are learned from the matching, and, finally, the tracking module is used for matching to achieve object tracking. Zhu et al. [92] introduced the ViT model to the field of multi-object tracking and proposed ViTT. It uses a transformer encoder as a backbone to extract features directly with the image as input. Compared to convolutional networks, it can model the global context at each encoder from the beginning and performs well in the challenges of occlusion and complex scenes. Yang et al. [93] consider the object position prediction and propose a transformer-based multi-object motion model, which takes the object's historical position difference and the offset vector between consecutive frames as input, and considers the object at the same time. The motion of itself and the camera improves the prediction accuracy of the motion model in the multi-object tracking method, thereby improving the tracking performance.

The transformer architecture in the neural network is intended to replace the traditional neural network that relies on global search, and to speed up and improve the efficiency of model training by instructing the model to focus more computing power or the gradient update of the parameters where we want it to, while ignoring irrelevant information in other areas. Most of the benchmark datasets have performed better than the conventional CNN architecture tracking technique. This demonstrates that research into algorithms with this architecture should continue [91].

6.2. Comparison of Transformer-Based MOT Algorithms

We provided a thorough synopsis of the transformer-based multi-object tracking algorithm in the previous section. Table 7 compares the fundamental concepts of the algorithm and the encoder–decoder structure.

The four approaches are not similar. The multi-object tracking method of the preceding transformer structure has been enhanced by TransCenter. According to the author, dealing with dense scenes is not a good fit for tracking methods based on the bounding boxes of the first two algorithms. Therefore, it is suggested to use the object heat map based on the center point to represent the image features, improving the handling of occlusion issues. TransCenter can also globally predict the center points of each object and associate them in the time domain, showing a higher FP–FN balance, thanks to the global characteristics of the transformer. In order to further improve the accuracy and precision of multi-object tracking, TransMOT believes that the present transformer-based methods are ineffective at modeling the spatiotemporal relationship. As a result, a series of sparse weighted graphs are created for modeling.

The transformer-based multi-object tracking method is still in its early stages. The transformer has an advantage over CNN as a detector since it employs self-attention to gather global contextual data, build remote dependencies on embedding, and extract more potent semantic characteristics. The transformer does, however, have certain drawbacks as well: transformer-based models need huge training datasets to work well, and the tracking performance for small objects is subpar since the self-attention operator's computational complexity rises exponentially as the embedding vector increases and the size of the patches

is constrained. In order to expand the framework to more difficult circumstances, such as multi-scale items, extreme weather conditions, category confusion, etc., the transformer-based multi-object tracking method needs further research.

Table 7. Algorithm analysis and comparison based on transformer-based framework.

Ref.	Method	Dataset	Basic Idea	Encoder-Decoder Architecture
[87]	TransTrack	MOT17 [25], MOT20 [26]	1. Using the Query–Key mechanism, the features of two adjacent frames are combined. 2. The object query is used to learn to detect new objects. 3. The track query is responsible for maintaining the trajectory. One of the two decoders is responsible for detection and generates a detection box; the other is responsible for object propagation and generates a tracking box. 4. Two boxes get the result by IoU matching.	Dual decoder
[88]	TrackFormer	MOT17 [25], MOT20 [26]	1. Using the Query–Key mechanism, the object query detects new objects, and the track query is responsible for tracking the object between frames (same as TransTrack) 2. In the decoder, the track queries and object queries are spliced and input, and the MLP directly maps the output of the decoder into classes and boxes.	Single decoder has position encoding and object encoding
[89]	TransCenter	MOT17 [25], MOT20 [26]	1. Siamese network structure, two adjacent frames are used as input. 2. One network is used for detection, the other network is used for tracking, and multi-scale features are generated for fusion, respectively, entering three branches to perform three tasks: predicting the center heat map, predicting the size of the box, and predicting the displacement.	Siamese Transformer
[90]	TransMOT	MOT15 [32], MOT16 [25], MOT20 [26]	1. The spatial relationship between objects is represented by a sparse weighted graph. 2. The set of graphs from the past few frames enters the Encoder to learn the temporal spatial relationship; the graph of the current frame enters the Decoder to learn the spatial relationship, and finally generates an allocation matrix for matching.	Graph Transformer

7. Experiment Analysis

In order to compare and analyze the current visual multi-object tracking algorithms based on deep learning, this chapter compares the performance of the algorithms through the MOT17 test set. Combined with the following conditions, the design experimental analysis table is shown in Tables 8 and 9. As introduced in Chapter 2, the larger the HOTA, MOTA, and IDF1 values, the better, and the smaller the IDs, FN, and FP, the better.

Table 8. In the MOT17 dataset [25], the algorithm evaluation index based on the public detector.

Framework	Ref.	Method	Year	HOTA	MOTA	IDF1	IDs	FPs
TBD	[50]	SORT	2016	34.0	43.1	39.8	4842	143.3
	[65]	MHT_bLSTM	2018	41.0	47.5	51.9	2069	1.9
	[57]	eHAF17	2018	—	51.8	54.7	1843	0.7
	[69]	OCSORT	2021	52.4	58.2	65.1	784	28.6
	[68]	BLSTM_MTP	2021	41.3	51.5	54.9	2566	20.1
JDT	[70]	Tracktor++	2019	42.1	53.5	52.3	2072	1.5
	[75]	JDE	2020	—	62.1	56.9	1608	30.3
	[73]	CenterTrack	2020	52.2	67.8	64.7	3039	3.8
	[72]	MIF	2020	—	60.1	56.4	2556	7.2
Transformer-based	[87]	TransTrack	2021	—	74.5	63.9	3663	—
	[88]	TrackFormer	2021	—	65.0	63.9	3528	—
	[89]	TransCenter	2021	51.4	68.2	61.4	4102	1.0
	[90]	TransMOT	2021	—	68.7	72.2	2346	20.1

Table 9. In the MOT17 dataset [25], the algorithm evaluation index based on the private detector.

Framework	Ref.	Method	Year.	HOTA	MOTA	IDF1	IDs	FPs
TBD	[51]	DeepSORT	2017	54.4	60.3	61.2	2821	20.8
	[59]	StrongSORT	2022	64.4	79.6	79.5	1194	7.1
JDT	[71]	FFT	2020	—	56.5	51.0	5672	—
	[74]	CTracker	2020	—	66.6	57.4	5529	34.4
	[12]	FairMOT	2021	59.3	73.7	72.3	3303	23
Transformer-based	[87]	TransTrack	2021	54.1	75.2	63.5	3603	59.2
	[88]	TrackFormer	2021	57.3	74.1	68.0	2829	5.7
	[89]	TransCenter	2021	54.5	73.2	62.2	4614	1.0

1. Among the above-mentioned algorithms, the most commonly used is the MOT17 test set. Therefore, in order to maintain fairness, we use the MOT17 data set test indicators to analyze the algorithm.
2. Due to the differences between public detection-based and private detection-based algorithms, the algorithm performance is discussed separately.
3. To highlight the difference between the three tracking algorithm frameworks, we still order these algorithms according to the three frameworks of TBD, JDT, and transformer-based, separated by line segments in the table.

7.1. Overall Analysis

The index scores of the multi-object tracking algorithms of the three frameworks tested in the MOT17 data set are listed in Tables 6 and 7. The overall analysis is as follows according to the test results:

1. The assessment indicators MOTA and IDF1, based on private detectors, are improved by 9.1 and 4.1, respectively, from the test indicators, based on public detectors, such as the transformer-based TrackFormer algorithm, when compared with public detectors. Therefore, the effectiveness of the front-end detector has a significant impact on multi-object tracking. Enhancing the detector's performance is essential for enhancing multi-object tracking's performance.
2. From the test index MOTA, the TBD framework, JDT framework, and transformer-based framework gradually improve the accuracy of multi-object tracking, the transformer-based algorithm especially constantly refreshes the current optimal performance on the dataset, showing the potential of transformer tracking.
3. The performance of IDs is a crucial indicator of the evaluation algorithm in practical applications, and the handling of the occlusion problem is a problem that must be taken seriously in traffic scenarios, as can be seen from Equation (1), which shows that

the IDs index has a relatively small impact on MOTA. According to the information provided above, several tracking algorithms are designed with the goal of reducing IDs and significantly enhancing MOTA; yet, relatively speaking, the performance of the most sophisticated tracking algorithms on IDs indicators is still subpar.

4. Considering that the majority of the challenges use the best hardware for algorithm testing at the moment, the FPs index is constrained by the inconsistency of the tested hardware and software platforms and cannot be used as an absolute algorithm speed assessment. As a result, the majority of the multi-object tracking algorithms used today have trouble keeping up with real-time multi-object tracking.

7.2. Advantages and Disadvantages of Multi-Object Tracking Algorithm Based on Deep Learning

We evaluated and analyzed the multi-target tracking method using the data set test results as an entry point in the previous section. The advantages and disadvantages of the TBD, JDT, and Transformer-based tracking algorithms are then briefly discussed.

(1) TBD Framework

The deep learning-based TBD algorithm continues to be the widely used research framework for multi-object tracking currently. The TBD algorithm has a straightforward layout and excellent interpretability. Modules such object identification, feature extraction, and data association can be employed by embedding deep learning modules in conventional modules. Tracking is stronger. The TBD algorithm's design is bloated, though, and it is challenging to strike a balance between accuracy and real-time performance.

(2) JDT Framework

The multi-module joint learning integration JDT algorithm has been created. The design structure is made simpler through the integration of several sub-modules, and an end-to-end multi-object tracking algorithm is produced, in which the detector performance is crucial to the effectiveness of the JDT algorithm. Currently, the multi-object tracking algorithm is typically moving from the TBD method to the JDT algorithm, and the algorithm's accuracy and speed are gradually balanced. In situations like frequent occlusion of numerous objects and having too many objects in the MOT dataset test, the JDT algorithm still performs inaccurately.

(3) Transformer-based Framework

Table 7 shows that TransTrack, TransCenter, and TransMOT, three multi-object tracking algorithms based on transformers, perform at the cutting edge on a variety of benchmark datasets (MOT16, MOT17, and MOT20). However, we think that tracking algorithms based on transformers still have a lot of drawbacks. First, the model for the transformer-based multi-object tracking algorithm is enormous and expensive to compute. For instance, TrackFormer's model size is 2.1 G. The FairMOT algorithm model, which is based on JDT, may, in contrast, achieve comparable performance with a 265 M file size. Due to this, the multi-object tracking algorithm's transformer architecture still adheres to the NLP standard transformer and has not been updated for the CV area.

8. Conclusions and Future Research

The visual multi-object tracking algorithms used in autonomous driving scenarios are thoroughly reviewed in this work. The algorithms are grouped into three categories: TBD, JDT, and transformer-based tracking, depending on their various structural configurations. Transformer-based algorithms have received a lot of attention over the last two years, and they offer a lot of room for further study in the multi-object tracking area. The theoretical framework for algorithmic study by experts in related domains is provided in this publication. Additionally, the approach we discuss, which uses merely monocular cameras rather than sophisticated sensor fusion, is anticipated to pave the way for the quick creation of safe and affordable autonomous driving systems. The possible future work includes the following aspects:

1. We seek a fundamental breakthrough in the use of multi-object tracking algorithms to solve real-world issues. We consider conducting research on novel network architectures, training techniques, cost functions, etc., to address issues such as the frequent IDs and inaccurate dense multi-object tracking. In addition, our future study focus will be on image feature extraction in various surroundings under various weather circumstances, such as too strong and too dark light, heavy fog, and severe rain.
2. The transformer structure is still based on the transformer used by the NLP business. The next stage is to create a transformer structure suitable for the CV field and a lightweight transformer model in order to balance the tracking algorithm's accuracy and speed.
3. The deployment of the algorithm to automobiles has to be improved. On the one hand, hardware platform adaptability optimization and hardware acceleration enhance the algorithm's real-time performance. On the other hand, thorough research into transfer learning, reinforcement learning, and other techniques to lessen the algorithm's reliance on data sets, enhances the algorithm's capacity for generalization.

Author Contributions: Conceptualization, S.G. and S.W.; methodology, Z.Y.; validation, L.W., H.Z. and P.G.; formal analysis, S.G.; data curation, S.W.; writing—original draft preparation, S.W.; writing—review and editing, S.G., Y.G.; visualization, J.G.; supervision, S.G.; project administration, S.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Ministry of education Industry–University Cooperation Education project, grant number 201901189017, Key scientific research projects of colleges and universities in Henan Province, grant number 23A470012, Scientific and Technological Innovation Team Support Program for Colleges and Universities in Henan Province, grant number 19IRTSTHN011 and Key scientific research project plan of colleges and universities in Henan Province, grant number 20A470008.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data is contained within the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Fagnant, D.J.; Kockelman, K. Preparing a nation for autonomous vehicles: Opportunities, barriers and policy recommendations. *Transp. Res. Part A Policy Pract.* **2015**, *77*, 167–181. [CrossRef]
2. Hussain, R.; Zeadally, S. Autonomous cars: Research results, issues, and future challenges. *IEEE Commun. Surv. Tutor.* **2018**, *21*, 1275–1313. [CrossRef]
3. Leon, F.; Gavrilescu, M. A Review of Tracking, Prediction and Decision Making Methods for Autonomous Driving. *arXiv* **2019**, arXiv:1909.07707v1.
4. Fan, L.; Wang, Z.; Cail, B.; Tao, C.; Feng, Z. A survey on multiple object tracking algorithm. In Proceedings of the 2016 IEEE International Conference on Information and Automation (ICIA), Ningbo, China, 1–3 August 2016.
5. Luo, W.; Xing, J.; Zhang, X.; Zhao, X.; Kim, T.K. Multiple Object Tracking: A Literature Review. *arXiv* **2014**, arXiv:1409.7618. [CrossRef]
6. Ciaparrone, G.; Sánchez, F.; Tabik, S.; Troiano, L.; Herrera, F. Deep Learning in Video Multi-Object Tracking: A Survey. *Neurocomputing* **2020**, *381*, 61–88. [CrossRef]
7. Sun, Z.; Chen, J.; Chao, L.; Ruan, W.; Mukherjee, M. A Survey of Multiple Pedestrian Tracking Based on Tracking-by-Detection Framework. *IEEE Trans. Circuits Syst. Video Technol.* **2020**, *31*, 1819–1833. [CrossRef]
8. Krebs, S.; Duraisamy, B.; Flohr, F. A survey on leveraging deep neural networks for object tracking. In Proceedings of the 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), Yokohama, Japan, 16–19 October 2017.
9. Wu, X.; Li, W.; Hong, D.; Tao, R.; Du, Q. Deep learning for unmanned aerial vehicle-based object detection and tracking: A survey. *IEEE Geosci. Remote Sens. Mag.* **2021**, *10*, 91–124. [CrossRef]
10. Kitchenham, B.; Brereton, O.P.; Budgen, D.; Turner, M.; Bailey, J.; Linkman, S. Systematic literature reviews in software engineering–a systematic literature review. *Inf. Softw. Technol.* **2009**, *51*, 7–15. [CrossRef]
11. Feichtenhofer, C.; Pinz, A.; Zisserman, A. Detect to track and track to detect. In Proceedings of the IEEE International Conference on Computer Vision (ICCV) 2017, Venice, Italy, 22–29 October 2017; pp. 3038–3046.
12. Zhang, Y.; Wang, C.; Wang, X.; Zeng, W.; Liu, W. Fairmot: On the fairness of detection and re-identification in multiple object tracking. *Int. J. Comput. Vis.* **2021**, *129*, 3069–3087. [CrossRef]

13. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Houlsby, N. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv* **2020**, arXiv:2010.11929v2.
14. Wang, X.; Girshick, R.; Gupta, A.; He, K. Non-local Neural Networks. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
15. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-end object detection with transformers. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020.
16. Mnih, V.; Heess, N.; Graves, A.; Kavukcuoglu, K. Recurrent models of visual attention. In Proceedings of the NIPS 2014, Montreal, QC, Canada, 8–13 December 2014.
17. Kim, C.; Li, F.; Ciptadi, A.; Rehg, J.M. Multiple hypothesis tracking revisited. In Proceedings of the IEEE International Conference on Computer Vision 2015, Santiago, Chile, 7–13 December 2015; pp. 4696–4704.
18. Davey, S.J.; Rutten, M.G.; Gordon, N.J. Track-before-detect techniques. In *Integrated Tracking, Classification, and Sensor Management*; Wiley Online Library: Hoboken, NJ, USA, 2013; pp. 311–362.
19. Wang, N.; Zhou, W.; Li, H. Reliable re-detection for long-term tracking. *IEEE Trans. Circuits Syst. Video Technol.* **2018**, *29*, 730–743. [CrossRef]
20. Pang, B.; Li, Y.; Zhang, Y.; Li, M.; Lu, C. Tubetk: Adopting tubes to track multi-object in a one-step training model. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2020, Seattle, WA, USA, 13–19 June 2020; pp. 6308–6318.
21. Ke, B.; Zheng, H.; Chen, L.; Yan, Z.; Li, Y. Multi-object tracking by joint detection and identification learning. *Neural Process. Lett.* **2019**, *50*, 283–296. [CrossRef]
22. Fortin, B.; Lherbier, R.; Noyer, J.-C. A model-based joint detection and tracking approach for multi-vehicle tracking with lidar sensor. *IEEE Trans. Intell. Transp. Syst.* **2015**, *16*, 1883–1895. [CrossRef]
23. Zeng, F.; Dong, B.; Wang, T.; Zhang, X.; Wei, Y. Motr: End-to-end multiple-object tracking with transformer. *arXiv* **2021**, arXiv:2105.03247.
24. Yu, E.; Li, Z.; Han, S.; Wang, H. Relationtrack: Relation-aware multiple object tracking with decoupled representation. *IEEE Trans. Multimed.* 2022. [CrossRef]
25. Milan, A.; Leal-Taixe, L.; Reid, I.; Roth, S.; Schindler, K. MOT16: A Benchmark for Multi-Object Tracking. *arXiv* **2016**, arXiv:1603.00831v2.
26. Dendorfer, P.; Rezatofighi, H.; Milan, A.; Shi, J.; Cremers, D.; Reid, I.; Roth, S.; Schindler, K.; Leal-Taixé, L. MOT20: A benchmark for multi object tracking in crowded scenes. *arXiv* **2020**, arXiv:2003.09003v1.
27. Geiger, A.; Lenz, P.; Urtasun, R. Are we ready for autonomous driving? The KITTI vision benchmark suite. In Proceedings of the IEEE Conference on Computer Vision & Pattern Recognition, Providence, RI, USA, 16–21 June 2012.
28. Geiger, A.; Lenz, P.; Stiller, C.; Urtasun, R. Vision meets robotics: The KITTI dataset. *Int. J. Robot. Res.* **2013**, *32*, 1231–1237. [CrossRef]
29. Caesar, H.; Bankiti, V.; Lang, A.H.; Vora, S.; Liong, V.E.; Xu, Q.; Krishnan, A.; Pan, Y.; Baldan, G.; Beijbom, O. nuscenes: A multimodal dataset for autonomous driving. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11621–11631.
30. Sun, P.; Kretzschmar, H.; Dotiwalla, X.; Chouard, A.; Patnaik, V.; Tsui, P.; Guo, J.; Zhou, Y.; Chai, Y.; Caine, B. Scalability in perception for autonomous driving: Waymo open dataset. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 2446–2454.
31. Ertler, C.; Mislej, J.; Ollmann, T.; Porzi, L.; Neuhold, G.; Kuang, Y. The mapillary traffic sign dataset for detection and classification on a global scale. In Proceedings of the ECCV 2020, Glasgow, UK, 23–28 August 2020; pp. 68–84.
32. Leal-Taixé, L.; Milan, A.; Reid, I.; Roth, S.; Schindler, K. Motchallenge 2015: Towards a benchmark for multi-target tracking. *arXiv* **2015**, arXiv:1504.01942.
33. Keni, B.; Rainer, S. Evaluating multiple object tracking performance: The CLEAR MOT metrics. *EURASIP J. Image Video Process.* **2008**, *2008*, 246309.
34. Luiten, J.; Osep, A.; Dendorfer, P.; Torr, P.; Geiger, A.; Leal-Taixé, L.; Leibe, B. Hota: A higher order metric for evaluating multi-object tracking. *Int. J. Comput. Vis.* **2021**, *129*, 548–578. [CrossRef] [PubMed]
35. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
36. Lowe, D.G. Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [CrossRef]
37. Dalal, N. Histograms of oriented gradients for human detection. In Proceedings of the Computer Vision and Pattern Recognition, San Diego, CA, USA, 20–26 June 2005; pp. 886–893.
38. Dai, J.; Li, Y.; He, K.; Sun, J. R-FCN: Object Detection via Region-based Fully Convolutional Networks. *arXiv* **2016**, arXiv:1605.06409v2.
39. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single shot multibox detector. In *Lecture Notes in Computer Science*; Springer: Berlin/Heidelberg, Germany, 2016.
40. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef] [PubMed]

41. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
42. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6517–6525.
43. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767v1.
44. Bochkovskiy, A.; Wang, C.Y.; Liao, H. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv* **2020**, arXiv:2004.10934v1.
45. Fortmann, T.; Bar-Shalom, Y.; Scheffe, M. Sonar tracking of multiple targets using joint probabilistic data association. *IEEE J. Ocean. Eng.* **2003**, *8*, 173–184. [CrossRef]
46. Blackman, S.S. Multiple hypothesis tracking for multiple target tracking. *IEEE Aerosp. Electron. Syst. Mag.* **2009**, *19*, 5–18. [CrossRef]
47. Jin, S.; Ma, X.; Han, Z.; Wu, Y.; Yang, W.; Liu, W.; Qian, C.; Ouyang, W. Towards multi-person pose tracking: Bottom-up and top-down methods. In Proceedings of the ICCV PoseTrack Workshop, Venice, Italy, 22–29 October 2017; p. 7.
48. Zhao, D.; Fu, H.; Xiao, L.; Wu, T.; Dai, B. Multi-Object Tracking with Correlation Filter for Autonomous Vehicle. *Sensors* **2018**, *18*, 2004. [CrossRef]
49. Li, T.; Xu, D.; Ma, Y.; Yu, C. A multiple object tracking algorithm based on YOLO detection. In Proceedings of the 2018 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), Beijing, China, 13–15 October 2018.
50. Bewley, A.; Ge, Z.; Ott, L.; Ramos, F.; Upcroft, B. Simple online and realtime tracking. In Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 25–28 September 2016; pp. 3464–3468.
51. Wojke, N.; Bewley, A.; Paulus, D. Simple online and realtime tracking with a deep association metric. In Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 17–20 September 2017.
52. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
53. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
54. Xiao, T.; Li, H.; Ouyang, W.; Wang, X. Learning deep feature representations with domain guided dropout for person re-identification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1249–1258.
55. Mahmoudi, N.; Ahadi, S.M.; Rahmati, M. Multi-target tracking using CNN-based features: CNNMTT. *Multimed. Tools Appl.* **2019**, *78*, 7077–7096. [CrossRef]
56. Fang, K.; Xiang, Y.; Li, X.; Savarese, S. Recurrent autoregressive networks for online multi-object tracking. In Proceedings of the 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Tahoe, NV, USA, 12–15 March 2018; pp. 466–475.
57. Sheng, H.; Zhang, Y.; Chen, J.; Xiong, Z.; Zhang, J. Heterogeneous association graph fusion for target association in multiple object tracking. *IEEE Trans. Circuits Syst. Video Technol.* **2018**, *29*, 3269–3280. [CrossRef]
58. Lin, X.; Li, C.T.; Sanchez, V.; Maple, C. On the detection-to-track association for online multi-object tracking. *Pattern Recognit. Lett.* **2021**, *146*, 200–207. [CrossRef]
59. Du, Y.; Song, Y.; Yang, B.; Zhao, Y. StrongSORT: Make DeepSORT Great Again. *arXiv* **2022**, arXiv:2202.13514v1.
60. Son, J.; Baek, M.; Cho, M.; Han, B. Multi-object tracking with quadruplet convolutional neural networks. In Proceedings of the Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition(CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 5620–5629.
61. Chandrasekar, K.S.; Geetha, P. Multiple objects tracking by a highly decisive three-frame differencing-combined-background subtraction method with GMPFM-GMPHD filters and VGG16-LSTM classifier. *J. Vis. Commun. Image Represent.* **2020**, *72*, 102905. [CrossRef]
62. Xiang, J.; Zhang, G.; Hou, J. Online multi-object tracking based on feature representation and Bayesian filtering within a deep learning architecture. *IEEE Access* **2019**, *7*, 27923–27935. [CrossRef]
63. Farhodov, X.; Moon, K.-S.; Lee, S.-H.; Kwon, K.-R. LSTM network with tracking association for multi-object tracking. *J. Korea Multimed. Soc.* **2020**, *23*, 1236–1249.
64. Milan, A.; Rezatofighi, S.H.; Dick, A.R.; Reid, I.D.; Schindler, K. Online multi-target tracking using recurrent neural networks. *Proc. Conf. AAAI Artif. Intell.* **2017**, *31*. [CrossRef]
65. Kim, C.; Li, F.; Rehg, J.M. Multi-object tracking with neural gating using bilinear lstm. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 200–215.
66. Ran, N.; Kong, L.; Wang, Y.; Liu, Q. A robust multi-athlete tracking algorithm by exploiting discriminant features and long-term dependencies. In Proceedings of the International Conference on Multimedia Modeling, Thessaloniki, Greece, 8–11 January 2019; pp. 411–423.
67. Lee, B.; Erdenee, E.; Jin, S.; Nam, M.Y.; Jung, Y.G.; Rhee, P.K. Multi-class multi-object tracking using changing point detection. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–10, 15–16 October 2016; pp. 68–83.

68. Kim, C.; Li, F.; Alotaibi, M.; Rehg, J.M. Discriminative Appearance Modeling with Multi-track Pooling for Real-time Multi-object Tracking. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021.
69. Cao, J.; Weng, X.; Khirodkar, R.; Pang, J.; Kitani, K. Observation-Centric SORT: Rethinking SORT for Robust Multi-Object Tracking. *arXiv* **2022**, arXiv:2203.14360.
70. Bergmann, P.; Meinhardt, T.; Leal-Taixe, L. Tracking without bells and whistles. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27 October 2019–02 November 2019.
71. Zhang, J.; Zhou, S.; Chang, X.; Wan, F.; Huang, D. Multiple Object Tracking by Flowing and Fusing. *arXiv* **2020**, arXiv:2001.11180v1.
72. Huang, P.; Han, S.; Zhao, J.; Liu, D.; Wang, H.; Yu, E.; Kot, C.C. Refinements in Motion and Appearance for Online Multi-Object Tracking. *arXiv* **2020**, arXiv:2003.07177.
73. Zhou, X.; Koltun, V.; Krhenbühl, P. Tracking Objects as Points. *arXiv* **2020**, arXiv:2004.01177v2.
74. Peng, J.; Wang, C.; Wan, F.; Wu, Y.; Fu, Y. Chained-Tracker: Chaining Paired Attentive Regression Results for End-to-End Joint Multiple-Object Detection and Tracking. *arXiv* **2020**, arXiv:2007.14557v1.
75. Wang, Z.; Zheng, L.; Liu, Y.; Li, Y.; Wang, S. Towards real-time multi-object tracking. *arXiv* **2020**, arXiv:1909.12605v2.
76. Zhu, J.; Yang, H.; Liu, N.; Kim, M.; Zhang, W.; Yang, M.H. Online Multi-Object Tracking with Dual Matching Attention Networks. *arXiv* **2019**, arXiv:1902.00749v1.
77. Feng, W.; Hu, Z.; Wei, W.; Yan, J.; Ouyang, W. Multi-Object Tracking with Multiple Cues and Switcher-Aware Classification. *arXiv* **2019**, arXiv:1901.06129v1.
78. Chu, P.; Fan, H.; Tan, C.; Ling, H. Online Multi-Object Tracking With Instance-Aware Tracker and Dynamic Model Refreshment. In Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 7–11 January 2019; pp. 161–170.
79. Bahdanau, D.; Cho, K.; Bengio, Y. Neural Machine Translation by Jointly Learning to Align and Translate. *arXiv* **2014**, arXiv:1409.0473v7.
80. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
81. Han, K.; Wang, Y.; Chen, H.; Chen, X.; Guo, J.; Liu, Z.; Tang, Y.; Xiao, A.; Xu, C.; Xu, Y.; et al. A survey on visual transformer. *arXiv* **2020**, arXiv:2012.12556.
82. Zhu, X.; Su, W.; Lu, L.; Li, B.; Wang, X.; Dai, J. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv* **2020**, arXiv:2010.04159.
83. Fu, J.; Liu, J.; Tian, H.; Li, Y.; Bao, Y.; Fang, Z.; Lu, H. Dual Attention Network for Scene Segmentation. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019.
84. Jin, Y.; Han, D.; Ko, H. Trseg: Transformer for semantic segmentation. *Pattern Recognit. Lett.* **2021**, *148*, 29–35. [CrossRef]
85. Lu, D.; Xie, Q.; Xu, L.; Li, J. 3DCTN: 3D Convolution-Transformer Network for Point Cloud Classification. *arXiv* **2022**, arXiv:2203.00828v1. [CrossRef]
86. Guo, M.-H.; Cai, J.-X.; Liu, Z.-N.; Mu, T.-J.; Martin, R.R.; Hu, S.-M. Pct: Point cloud transformer. *Comput. Vis. Media* **2021**, *7*, 187–199. [CrossRef]
87. Sun, P.; Jiang, Y.; Zhang, R.; Xie, E.; Luo, P. TransTrack: Multiple-Object Tracking with Transformer. *arXiv* **2020**, arXiv:2012.15460v2.
88. Meinhardt, T.; Kirillov, A.; Leal-Taixe, L.; Feichtenhofer, C. TrackFormer: Multi-Object Tracking with Transformers. *arXiv* **2021**, arXiv:2101.02702v3.
89. Xu, Y.; Ban, Y.; Delorme, G.; Gan, C.; Rus, D.; Alameda-Pineda, X. TransCenter: Transformers with Dense Queries for Multiple-Object Tracking. *arXiv* **2021**, arXiv:2103.15145v4.
90. Chu, P.; Wang, J.; You, Q.; Ling, H.; Liu, Z. TransMOT: Spatial-Temporal Graph Transformer for Multiple Object Tracking. *arXiv* **2021**, arXiv:2104.00194v2.
91. Xie, F.; Wang, C.; Wang, G.; Yang, W.; Zeng, W. Learning Tracking Representations via Dual-Branch Fully Transformer Networks. *arXiv* **2021**, arXiv:2112.02571v1.
92. Zhu, X.; Jia, Y.; Jian, S.; Gu, L.; Pu, Z. ViTT: Vision Transformer Tracker. *Sensors* **2021**, *21*, 5608. [CrossRef]
93. Yang, J.; Ge, H.; Su, S.; Liu, G. Transformer-based two-source motion model for multi-object tracking. *Appl. Intell.* **2022**, *52*, 9967–9979. [CrossRef]