YOLO11-JDE: Fast and Accurate Multi-Object Tracking with Self-Supervised Re-ID

Iñaki Erregue[1] ierregal31@alumnes.ub.edu
Kamal Nasrollahi[3,4] kn@create.aau.dk
Sergio Escalera[1,2,3] sescalera@ub.edu

[1]Universitat de Barcelona [2]Computer Vision Center [3]Aalborg University [4]Milestone Systems

Abstract

We introduce YOLO11-JDE, a fast and accurate multi-object tracking (MOT) solution that combines real-time object detection with self-supervised Re-Identification (Re-ID). By incorporating a dedicated Re-ID branch into YOLO11s, our model performs Joint Detection and Embedding (JDE), generating appearance features for each detection. The Re-ID branch is trained in a fully self-supervised setting while simultaneously training for detection, eliminating the need for costly identity-labeled datasets. The triplet loss, with hard positive and semi-hard negative mining strategies, is used for learning discriminative embeddings. Data association is enhanced with a custom tracking implementation that successfully integrates motion, appearance, and location cues. YOLO11-JDE achieves competitive results on MOT17 and MOT20 benchmarks, surpassing existing JDE methods in terms of FPS and using up to ten times fewer parameters. Thus, making our method a highly attractive solution for real-world applications. The code is publicly available at https://github.com/inakierregueab/YOLO11-JDE.

1. Introduction

Multi-Object Tracking (MOT) is a fundamental task in computer vision that involves detecting multiple objects in a video sequence and maintaining their identities discriminated across frames. From autonomous driving [8, 24, 38] and video surveillance [2, 51] to sports analytics [12, 29, 53] and robotics [42, 58], MOT is a key component for numerous real-world applications. Despite significant advancements in the field, factors such as frequent occlusions, complex and unpredictable motion patterns, and the need for real-time performance in practical scenarios remain challenging [3, 11].

Among the different paradigms for MOT, the Tracking-by-Detection (TbD) approach has become the most widely used due to its modularity and flexibility, separating the task into two stages: detecting objects in each frame and associating these detections across consecutive frames to maintain identities. Many methods integrate Re-Identification (Re-ID) embeddings to ease the matching process. These appearance cues are particularly valuable in challenging scenarios involving occlusions or objects with similar motion patterns, as they provide an additional layer of discrimination beyond spatial and temporal information.

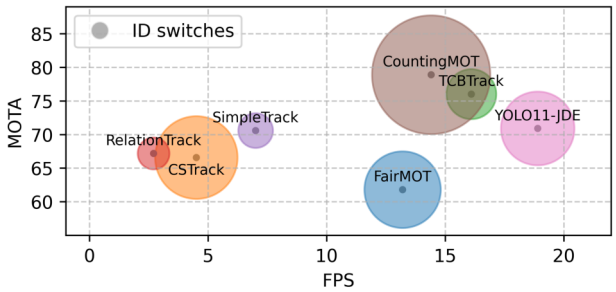

Figure 1. Comparative analysis of JDE models on the MOT20 test set. MOTA and FPS on the vertical and horizontal axis respectively. ID Switches are represented by the bubble size. YOLO11-JDE achieves a strong balance between tracking performance and inference speed.

While substantial progress has been made in both detection and Re-ID fields, most methods adopt a two-stage approach, known as Separate Detection and Embedding (SDE), where detection and Re-ID are performed independently [1, 28, 50, 56]. While effective, these methods suffer from scalability issues due to the lack of feature sharing and the computational cost of applying the Re-ID model to every bounding box. To address these limitations, recent advancements introduced Joint Detection and Embedding (JDE) models, which unify object detection and Re-ID feature extraction processes into a single model [18, 32–34, 45, 57, 63, 67, 69]. By sharing features between the two tasks and jointly optimizing them, JDE models significantly reduce computational overhead, making it an attractive paradigm for MOT.

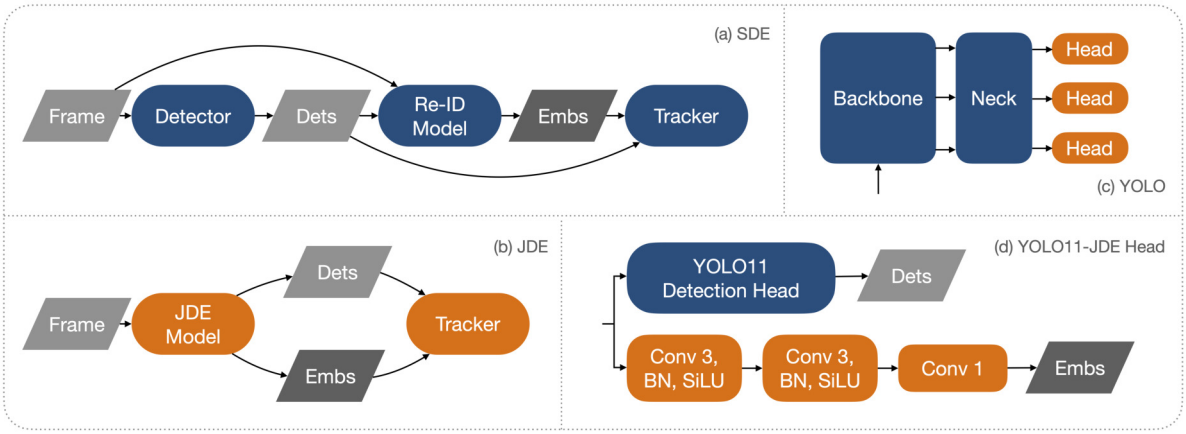

Figure 2. Comparison of tracking architectures: (a) Separate Detection and Embedding (SDE), where detection and Re-ID are performed by separate models; (b) Joint Detection and Embedding (JDE), integrating detection and Re-ID into a single model; (c) Basic YOLO model structure including the backbone, neck, and multiple output heads; and (d) YOLO11-JDE head, featuring a specialized Re-ID branch. Grey parallelograms represent data, while rounded rectangles depict models. Our main contributions are highlighted in orange.

Joint training of the detection and embedding tasks of JDE models pose unique challenges. While object detection focuses on clustering features to separate object classes, Re-ID requires some intra-class variability to achieve good discrimination between individual objects within the same class. This inherent conflict complicates the optimization process, making the choice of loss functions particularly critical in JDE models. Furthermore, achieving state-of-the-art performance often requires large-scale, labelled tracking datasets for supervision, which are expensive and time-consuming to obtain.

In this work, we present an end-to-end framework that builds upon the popular Ultralytics [27] framework and the state-of-the-art detector, YOLO11 [26], modified to perform joint detection and embedding. To tackle the inherent difficulties in joint training, we explore the field of deep metric learning, aiming to strike an optimal balance between detection and embedding objectives by using the well-established triplet loss [9, 48]. Moreover, to mitigate the need for extensive identity label supervision, we utilize strong data augmentation techniques, particularly Mosaic data augmentation [6], enabling our model to perform effectively in a fully self-supervised setting. Our approach drastically reduces the number of parameters compared to existing JDE methods, resulting in a notable speed-up in frames per second (FPS). Evaluated on the MOT Challenge benchmarks, YOLO11-JDE demonstrates competitive tracking accuracy while maintaining high efficiency (see Fig. 1), making it well-suited for real-time MOT applications, where inference speed and model size are crucial. In summary, our main contributions are:

- YOLO11-JDE, a modified YOLO11s that performs JDE, being small, fast and accurate.
- A self/semi-supervised setting for training JDE models based on Mosaic data augmentation and the triplet loss function.
- A customized data association algorithm that integrates motion, location and appearance cues.

2. Related Work

2.1. Tracking-by-Detection

The task of MOT can be broadly categorized into three main paradigms based on how detection and tracking tasks are combined: tracking-by-regression, tracking-by-detection and tracking-by-attention. Nonetheless, TbD stands out as the most practical and widely used approach in both research and real-world applications. These trackers divide MOT into two separate tasks: detection and association. The tracking process begins with the identification of potential objects of interest in each frame using high-performance detectors like YOLOX [20], Faster R-CNN [44], or CenterNet [71]. Detected objects are then associated across consecutive frames using tracker algorithms that perform data association employing several cues (motion, location, appearance, etc.).

Since the candidate boxes can be directly provided by off-the-shelf detectors, TbD methods mainly focus on improving the association performance. Early methods like SORT [5] employ a Kalman filter [30] to predict object positions in subsequent frames, assuming linear motion dynamics. Data association is performed using the Hungarian algorithm [31], with a cost matrix based on the Intersection-over-Union (IoU) between predicted and detected bounding boxes. More recent advancements, like ByteTrack [68], utilize all outputted detections, including low-confidence ones,

in a two-stage cascade matching strategy. ConfTrack [28] and BoostTrack [50] go one step further by introducing novel penalization and boosting methods for low- and high-confidence detections in the matching process respectively. In a different direction, C-BIoU tracker [61] mitigates the effect of irregular motions by adding buffers to expand the matching space of detections and tracks.

2.2. Re-Identification

To better deal with occlusions, crowded scenes, and non-linear motion, appearance similarity is commonly used in addition to IoU and motion cues. Thus, modern systems, like DeepSORT [59], BoT-SORT [1], SMILETrack [56] and many others [28, 50, 64], incorporate the extraction of discriminative Re-ID features for detected objects. These embeddings can be obtained either using an external high-quality feature extractor (*e.g*. FastReID [22]), or using JDE models (see Figs. 2a and 2b). Despite achieving superior performance, SDE approaches suffer from massive computation costs since the feature extractor network needs to perform forward inference on the image or feature map crop of each bounding box, thus limiting real-time applications.

2.3. Joint Detection and Embedding

JDE models perform object detection and Re-ID feature extraction in a single network in order to reduce inference time. Focusing on one-shot detectors, Wang *et al*. [57] redesign the coupled prediction head of YOLOv3 [43] to extract embeddings of dimension 512 directly applying a 1×1 convolution layer on the shared features. Thus, ignoring the inherent differences between the three tasks involved. Moreover, [57] trains the Re-ID task using a classification approach, where extracted embeddings are fed into a shared fully-connected layer to output the class-wise logits, and then cross-entropy loss is applied. In this method, annotations without identity labels are ignored. CSTrack [34] adopts YOLOv5 [25] as detector and introduces two new modules to decouple the Re-ID task and fuse embeddings across scales. Subsequent advancements, such as OMC [33] and TCBTrack [67], emphasize the temporal refinement of appearance cues.

On the other hand, FairMOT [69] uses a modified version of the anchor-free detector CenterNet to output 128-dimensional features alongside each detection. Similarly to previously mentioned approaches, FairMOT learns Re-ID features through a classification task. In addition to the standard training strategy, FairMOT introduces a single-image training approach tailored for image-level object detection datasets. Each bounding box is assigned a unique identity, effectively treating every object instance in the dataset as a distinct class. By applying various transformations to the entire image, the model is exposed to each identity across multiple conditions. Despite reporting acceptable results, this self-supervised approach is used as a pre-training routine and not explored any deeper. QDTrack [18] further investigates the self-supervised paradigm incorporating MixUp [65] and Mosaic transformations, along with an extension of the InfoNCE loss [52] paired with a regularization term. Meanwhile, other models based on CenterNet, like RelationTrack [63] and SimpleTrack [32] focus on decoupling both tasks and improving data association.

More recent JDE methods, CountingMOT [45] and UTM [62], achieve state-of-the-art performance on MOTChallenge benchmarks. The former, build upon FairMOT, adds an extra counting task to be shared across detection and density estimation branches, boosting the performance in crowded scenes. The latter, includes the data association step into a unified tracker model, creating a positive feedback loop boosting detection and Re-ID altogether.

Despite being designed for autonomous driving scenarios, RetinaTrack [38] is also noteworthy. Designed on top of RetinaNet [35], it performs the task of JDE using the triplet loss and mining hard triplets.

3. YOLO11-JDE

In this section we detail the technical aspects of YOLO11-JDE including its modified architecture, the different strategies employed for effectively training the Re-ID branch in a self-supervised fashion, and the integration of Re-ID embeddings into the online data association process.

3.1. Architecture

Following related JDE approaches like [18,34,57,62,67], our framework is based on the YOLO family of detectors, which typically consist of a backbone for generating feature maps, a neck that refines them by fusing shallow and deep representations, and three prediction heads (see Fig. 2c). Particularly, the state-of-the-art version YOLO11s has been chosen for its efficiency, accuracy and real-time performance. We have incorporated a Re-ID branch in the original multi-task decoupled head, taking inspiration from the design of the bounding box and segmentation regression branches. The Re-ID branch processes input feature maps through two consecutive 3x3 convolutional layers, each followed by batch normalization and the SiLU activation function. A third 1x1 convolutional layer maps the features into the corresponding embedding dimension with no batch normalization applied, following best practices suggested in [22]. This simple yet effective design allows the Re-ID branch to learn discriminative features without introducing unnecessary complexity and assessing the task in a consistent manner with the other object detection tasks: classification and bounding box regression. Thus, YOLO11-JDE outputs an appearance embedding for each detection alongside its predicted class and bounding box (see Fig. 2d).

Figure 3. Example of four training images using Mosaic data augmentation for JDE. This technique combines multiple images, showing several identities (*e.g.*, IDs 45543, 45544, 45549) under diverse transformations in a single input image and/or batch.

3.2. Self-Supervised Training Strategy

The goal of the Re-ID branch is to produce robust embeddings that facilitate data association between consecutive frames, while minimizing the reliance on large-scale labelled tracking datasets. To achieve this, we aim for a fully self-supervised training approach, inspired by the work of FairMOT and QDTrack.

A core aspect of our self-supervised strategy is the use of Mosaic data augmentation [6], a technique commonly employed in training modern object detectors like YOLO11. Mosaic augmentation works by combining four different image patches into a single input image, effectively enabling the model to review the same identities under diverse transformations, including variations in color, scale, rotation, etc. As depicted in Fig. 3, this approach allows the JDE model to learn robust features by exposing them to multiple augmented versions of the same identity within the same input image and/or batch while ordinarily training for detection. Thus, learning to output appearance features for each detection almost for free.

While our approach is intended to be fully self-supervised, it is also compatible with semi-supervised training, where a small amount of labelled tracking/identity data can complement the training procedure. This flexibility ensures the framework can adapt to scenarios with varying levels of data availability, which is crucial in real-world application.

3.3. Re-ID Loss

For a given training batch, the model outputs N foreground predictions, each with an associated embedding that has a ground truth identity label assigned. The objective of the loss function is to pull embeddings with the same identity (positives) close together in the feature space while pushing those of different identities (negatives) further apart. This learning paradigm is a central concept in deep metric learning, where the goal is to learn a feature space where distances directly encode meaningful relationships between data points. The Re-ID task can be approached either as a classification problem or directly optimizing pair-wise relative distances between embeddings [7].

Inspired by common Re-ID models trained in an contrastive fashion [10, 19, 21, 22, 37], we adopt a pair-wise approach given its scalability to a large numbers of identities. While advanced pair-wise losses like Multi-Similarity [55], InfoNCE [52], or Angular [54] offer enhanced performance in certain tasks, we chose the triplet loss due to its simplicity, efficiency, and proven effectiveness [23]. The triplet loss aims to enforce a margin m between positive and negative samples by ensuring that an anchor (a sample from a given identity) is closer to its positive counterpart than to a negative sample. The loss function is defined as:

$$L_{triplet} = \sum_{\{a,p,n\}} [d_{ap} - d_{an} + m]_+ , \tag{1}$$

where $\{a, p, n\}$ represents the set of all triplets to be evaluated; and d_{ap} and d_{an} represent the distances between the anchor and the positive and negative samples respectively.

Given N embeddings, the total number of triplets scales as $O(N^3)$. This rapid growth makes it computationally unfeasible to use all possible combinations directly. Moreover, many of the resulting triplets would offer little new information to the model, slowing down convergence. To address these issues, recent advancements in pair-based metric learning have focused on more informative sampling strategies. In our setup, we use hard positive and semi-hard negative sampling strategy to obtain a total of N triplets, although other strategies are also explored, Sec. 4.3.1. On the one hand, hard positive mining chooses as positive for each anchor its furthest embedding with the same identity (most dissimilar). On the other hand, semi-hard negative mining selects the hardest negative sample per anchor (most similar embedding with different identity), such that it is further than the selected positive. Thus, selecting negatives pairs that are not too easy (far apart) but also not too difficult (close together). By utilizing these sampling strategies, we ensure that each triplet is informative and challenging, accelerating convergence, improving the overall performance and mitigating the issues of computational infeasibility.

3.4. Data Association

Initially, we adopted the two-stage online data association strategy used in FairMOT. Tracklets are initialized from detections in the first frame and updated in subsequent frames using a combination of motion and appearance cues. In the first stage, a Kalman Filter predicts tracklet locations and the Mahalanobis distance is computed between predicted and detected boxes. Normalized Re-ID embeddings are use to compute a cosine distance matrix, which is fused with the Mahalanobis distance to obtain the final cost matrix. The matches are determined using the Hungarian algorithm. In the second stage, unmatched tracklets and detections are linked based on bounding box IoU with a stricter matching threshold. Unmatched detections can initialize new tracks, while unmatched tracklets persist for 30 frames to handle occlusions. Following [16], appearance features are updated using an exponential moving average.

Building on FairMOT's tracker and inspired by ByteTrack, we have implemented a simple yet effective custom tracker for the YOLO11-JDE model. In the first stage, confident predictions are matched using a combination of motion, appearance, and localization cues. Motion is fused with appearance, following the approach of FairMOT, while also discarding matches with low IoU overlap. The IoU distance matrix is then combined with the confidence scores of detections, and low-similarity matches are discarded. The final cost matrix is a linear combination of these two factors. For low-confidence predictions and unmatched detections, linkage is performed using IoU alone. This approach balances computational simplicity with robust tracking performance.

4. Experiments

4.1. Dataset and Metrics

Seven datasets are commonly used when training JDE models on pedestrian tracking. Detection datasets include CrowdHuman [49], ETH [17] and CityPersons [66], while MOT17 [13], CalTech [15], CUHK-SYSU [60] and PRW [70] also provide identity annotations. In our study we will only explore the mentioned object detection datasets with the exception of MOT17, which is added to fine-tune the model for the final evaluation. Following previous work [50,57], we construct a MOT17 validation set using the second half of each training sequence and chop off the videos in ETH that are overlapped with the MOT16 [40] benchmark.

We evaluate our approach on the testing sets of two widely recognized benchmarks, MOT17 and MOT20 [14]. For overall tracking accuracy, we primarily rely on HOTA [39] due to its balanced evaluation of detection, association and trajectory quality. However, we also consider IDF1 [46] and MOTA from CLEAR metrics [4] to provide additional insights into identity preservation and overall tracking performance. Detection performance is assessed using Average Precision (AP) with the common 50:95 acceptance IoU threshold range. While the quality of the Re-ID embeddings and the training convergence are monitored using clustering metrics like the Silhouette score [47], retrieval mean Average Precision, and simpler indicators like the mean positive and negative Euclidean and cosine distances.

4.2. Implementation Details

Our framework builds on the Ultralytics infrastructure, modified to handle the task of JDE by incorporating identity labels management, a new JDE head, metrics for monitoring joint optimization, and a new set of tracking algorithms. Additionally, JDE loss functions and mining strategies are implemented using the PyTorch Metric Learning library [41]. Identity annotations are processed from existing datasets or generated synthetically if not available. They are preserved during data augmentation and foreground prediction alignment. All experiments use the YOLO11s model with COCO [36] pre-trained weights. The default configuration of hyperparameters for optimization and data augmentation is used, except for Mosaic, which is applied throughout the whole training.

4.3. Ablative Studies

In this section we present rigorous studies of four critical factors in YOLO11-JDE, including the Re-ID loss, the dimensionality of the appearance features, and the amount of training data and supervision needed. A simplified experimental setup has been employed to isolate and analyze the impact of these factors while maintaining computational feasibility. Specifically, we adopt the small variant of YOLO11 as the baseline model, trained for 30 epochs with a batch size of 32. The Re-ID branch employs the triplet loss with a unitary weight and outputs 128-dimensional embeddings. Training data is limited to CrowdHuman [49] and the detections from the training half of MOT17, all resized to 640 pixels. For validation, detection performance is evaluated on the validation splits of both datasets, while the Re-ID performance is exclusively assessed using the ground truth identity labels from MOT17. The tracker algorithm from FairMOT, with its default configuration, is used for evaluating the ablations, including an inference resolution of 1088×608 pixels. To ensure a comprehensive evaluation and to account for potential interactions between factors, a sequential approach has been taken, where the best-performing configuration from one ablation is used as the baseline for the next. Evaluation metrics are given in percent. The best results of each ablation are shown in **bold**.

Mining Strategy (+,-)	HOTA	MOTA	IDF1
(Hard, Hard)	51.66	55.23	59.41
(Hard, Semi-hard)	**55.91**	56.04	**65.31**
(Hard, Easy)	51.01	56.22	58.76
(Semi-hard, Hard)	53.93	55.11	62.34
(Semi-hard, Easy)	50.46	**57.03**	57.90
(Easy, Hard)	53.76	54.21	61.50
(Easy, Semi-Hard)	55.72	56.76	65.25

Table 1. Ablation results comparing different mining strategies for the triplet loss, showing the impact on tracking performance and Re-ID embedding quality.

4.3.1 Re-ID Loss

Mining Strategy. Ablation experiments begin by selecting the best mining strategy for the triplet loss, which has been used with the default margin $m = 0.05$. Various mining strategies have been explored, incorporating hard, semi-hard, and easy pairs for both positives and negatives. The results, summarized in Tab. 1, indicate that hard positives and semi-hard negatives yields the best overall performance both in terms of tracking accuracy and the quality of the Re-ID embeddings. This is likely due to the balanced challenge it presents to the model. Semi-hard negatives, which are not overly difficult to separate, are crucial for refining the decision boundary without introducing training instability. Meanwhile, hard positives force the model to learn robust discriminative features, enhancing intra-class consistency. Easier strategies, specially for negatives, rarely violate the margin condition, causing the model to focus mainly on the detection task, *i.e.*, higher MOTA.

Loss Margin. The next set of experiments focuses on the impact of the margin value m in the triplet loss function. As shown in Tab. 2, several values were tested around the baseline, with $m = 0.075$ yielding to the best performance in terms of HOTA, MOTA and IDF1. Two additional experiments were conducted using this margin. First, swapping the distance computation (*i.e.*, using the positive-negative distance instead of the anchor-negative if the latter violates the margin more) downgraded the performance, likely because it weakens the impact of the mining strategy. Second, smoothing the loss function by replacing the Hinge function with the Softplus function led to a noticeable increase in detection performance, although it slightly lags behind in HOTA.

Confidence Filtering. Following the margin analysis, we investigate the impact of filtering the embeddings used for triplet mining, focusing on a confidence-based selection. The default approach mines among all available embeddings, ensuring maximum coverage but potentially including noisy or low-confidence samples. Therefore, we try limiting the embeddings to the top 75% and 50% most confident predictions per batch. The results, summarized in Tab. 3, illustrate how the model performs better when using all predictions. This could be attributed to the additional diversity provided by lower-confidence samples, which may expose the Re-ID branch to a wider range of challenging cases, ultimately leading to more robust feature learning.

Loss Margin (m)	HOTA	MOTA	IDF1
0.025	56.03	56.81	65.40
0.05	55.91	56.04	65.31
0.075	**56.37**	56.38	66.45
0.1	55.62	55.88	65.42
0.075 (smooth)	56.21	**57.61**	**66.50**
0.075 (swap)	55.42	56.07	64.76

Table 2. Ablation results comparing different margin values for the triplet loss, as well as its smoothed and swapping distance counterparts.

Confidence Filtering (%)	HOTA	MOTA	IDF1
100	**56.37**	56.38	**66.45**
75	55.88	56.52	65.56
50	55.64	**56.69**	64.34

Table 3. Ablation results comparing different confidence filtering thresholds during the mining process.

Loss Weight	HOTA	MOTA	IDF1
0.5	55.92	**56.59**	65.24
1	**56.37**	56.38	**66.45**
1.5	55.27	56.59	64.58

Table 4. Ablation results comparing different weight values for the triplet loss.

Loss Weight. The last set of ablations evaluates the impact of using three different weight values for the Re-ID loss. The goal is to understand how varying the contribution of the triplet loss in the overall multi-task objective function influences tracking performance. Table 4 displays how the unitary weight outperforms the other configurations. Across several experiments, we have observed a general tendency: the lower the magnitude of the loss function, the better the results in detection. This suggests that a low but efficient signal in the Re-ID loss is crucial for ensuring that the detection task is not harmed during joint training.

4.3.2 Feature Dimension

In this subsection, we investigate the effect of varying the dimensionality of the embedding features on joint optimization and the final tracking performance. By experimenting with dimensions of 64, 128, and 256, we aim to identify

Feat. Dim.	HOTA	MOTA	IDF1
64	56.27	**58.31**	66.03
128	**56.37**	56.38	**66.45**
256	55.44	56.26	65.09

Table 5. Ablation results on the Re-ID feature dimensionality.

Training Data	HOTA	MOTA	IDF1
CH	52.39	**58.53**	65.12
CH, MOT17*	**56.37**	56.38	**66.45**
CH, MOT17	55.21	57.06	64.54
CH, ETH, CP, MOT17*	56.20	57.38	65.96
CH, ETH, CP, MOT17	55.54	57.83	65.14

Table 6. Ablation results on the self/semi-supervised approaches. Superscript* means that no identity annotations are used.

the optimal size that provides robust identity embeddings while maintaining computational feasibility. As depicted in Tab. 5, size 128 achieves the best balance. Dimension of 64 produces the highest MOTA, likely due to the lower signal loss benefiting detection during joint training. Conversely, increasing the dimension to 256 leads to slight declines across all metrics, probably caused by overfitting or redundant information in the higher-dimensional space.

4.3.3 Training Datasets

To assess the effect of the different types of supervision and data used in the training of our JDE model, we conducted another set of experiments. As shown in Tab. 6, the model trained on CrowdHuman alone achieves a strong MOTA score, but its lower HOTA score reflects the necessity of fine-tuning with MOT17. Interestingly, incorporating identity supervision does not lead to improvements in HOTA or IDF1, suggesting that the model effectively learns more discriminative features using a fully self-supervised approach. While adding additional datasets such as ETH and CityPersons enhance detection performance, they do not improve tracking metrics, highlighting that the quality and relevance of fine-tuning data are more critical than data diversity.

4.4. Data Association

With the most promising configuration identified, we have trained the model for a 100 epochs using a batch size of 64 and an input image resolution of 1280 pixels. We then focus on fine-tuning the hyperparameters involved in the data association step. This section compares the results of using the default FairMOT tracker with its original parameters against a fine-tuned version for the YOLO11-JDE model, using the MOT17 training split. The default tracker, without adaptation, may struggle with mismatched confidence distributions and feature representations, leading to suboptimal data association and tracking accuracy. As shown in Tab. 7, fine-tuning the tracker to align with YOLO11-JDE's specific outputs significantly enhances its overall effectiveness. Moreover, the custom YOLO11-JDE tracker outperforms both FairMOT trackers across all metrics, ensuring more precise data association by integrating motion, appearance, and localization cues.

Tracker	HOTA	MOTA	IDF1
FairMOT	56.89	57.84	66.89
FairMOT (fine-tuned)	57.25	57.10	67.16
YOLO11-JDE	**60.06**	**58.25**	**71.84**

Table 7. Results using different data association algorithms.

4.5. Results on MOTChallenge

We compare our method to existing literature, focusing on online JDE models targeting real-time performance. During inference, we use the new YOLO11-JDE tracker at an input resolution of 1280 pixels. Results on MOT17 and MOT20 test sets under the private detection protocol are shown in Tab. 8. Despite being the only fully self-supervised method in the comparison, YOLO11-JDE demonstrates competitive performance across benchmarks and significantly outpaces its counterparts in terms of FPS. When it comes to identity switches (IDs), YOLO11-JDE outperforms many of its competitors, demonstrating the discriminative power of the produced embeddings. Therefore, we attribute its lower performance in overall tracking to the limitations of the model's detection capabilities, rather than its re-identification ability. Furthermore, YOLO11-JDE has less than 10M parameters, while top-performing methods like CountingMOT rely on computationally expensive detectors such as YOLOX-X (100M parameters) or CenterNet (22M parameters).

Interestingly, YOLO11-JDE performs better in MOT20 than in MOT17 when compared with its competitors. It is important to note that neither the YOLO11-JDE model nor the tracker have been trained using the MOT20 dataset. This improved performance on crowded scenes (see Fig. 4) can be attributed to the type of data used in training. The CrowdHuman dataset, which has a density of nearly 23 people per image, is magnified by Mosaic data augmentation, turning to approximately 90 people per image. This data composition makes YOLO11-JDE highly robust when handling crowded scenarios and partial occlusions.

5. Summary and Future Work

In this work, we presented YOLO11-JDE, a lightweight and efficient MOT framework built upon YOLO11s, equipped with a Re-ID branch for joint detection and em-

Tracker	Detector	MOT17					MOT20				
		HOTA	MOTA	IDF1	IDs	FPS	HOTA	MOTA	IDF1	IDs	FPS
JDE [57]	YOLOv3	-	63.0	59.5	-	18.8	-	-	-	-	
FairMOT [69]	CenterNet	59.3	73.7	72.3	3303	25.9	54.6	61.8	67.3	5243	13.2
CSTrack [34]	YOLOv5	59.3	74.9	72.3	3567	15.8	54.0	66.6	68.8	3196	4.5
OMC [33]	YOLOv5	-	76.3	73.8	-	12.8	-	70.7	67.8	-	6.7
TCBTrack [67]	YOLOX-X	62.1	79.3	75.8	2157	27.7	60.6	76.0	74.4	1174	16.1
RelationTrack [63]	CenterNet	61.0	73.8	74.7	**1374**	7.4	56.5	67.2	70.5	4243	2.7
SimpleTrack [32]	CenterNet	61.0	74.1	75.7	1500	22.5	56.6	70.6	69.6	2434	7.0
QDTrack [18]	YOLOX-X	63.5	78.7	77.5	1935	-	60.0	74.7	73.8	**1042**	-
CountingMOT [45]	YOLOX-X	**63.6**	**81.3**	**78.4**	5118	26.4	**63.6**	**78.9**	**78.6**	1232	14.4
YOLO11s-JDE	YOLO11s	56.6	65.8	70.3	3177	**35.9**	53.1	70.9	66.4	3091	**18.9**

Table 8. Comparison with the state-of-the-art JDE models under the private detection protocol on MOT17 and MOT20 benchmarks. The best results are displayed in **bold** while the second best are underlined.

Figure 4. Example of consistent identity maintenance across frames on the MOT20-04 sequence despite multiple occlusions (ID 248).

bedding. Our method demonstrates that Re-ID can be effectively trained in a fully self-supervised manner, avoiding the need for identity-labeled datasets while maintaining competitive tracking performance. By combining the triplet loss with hard positive and semi-hard negative mining strategies, YOLO11-JDE produces discriminative embeddings that are robust across various tracking scenarios, particularly crowded environments. Additionally, we developed a custom tracking algorithm that integrates motion, appearance, and location cues, effectively improving data association and aligning seamlessly with YOLO11-JDE's outputs. Evaluations on the MOT17 and MOT20 benchmarks highlight the method's ability to deliver comparable accuracy to state-of-the-art models while achieving superior FPS and using significantly fewer parameters. These qualities make YOLO11-JDE a practical and scalable solution for real-world applications.

For future work, we aim to address the limitations observed in detection performance by refining the architecture to better decouple Re-ID and detection tasks. Further improvements to appearance features, such as incorporating multi-scale embedding fusion, could enhance Re-ID robustness. Additionally, we plan to investigate the impact of stronger data augmentations, including rotations, shear and perspective transformations, Mixup and random patch erasing within bounding boxes.

Acknowledgements. This work has been partially supported by the Spanish project PID2022-136436NB-I00 and by ICREA under the ICREA Academia programme, and by the Milestone Research Program at the University of Barcelona.

References

[1] Nir Aharon, Roy Orfaig, and Ben-Zion Bobrovsky. Bot-sort: Robust associations multi-pedestrian tracking, 2022. 1, 3

[2] Imran Ahmed, Sadia Din, Gwanggil Jeon, Francesco Piccialli, and Giancarlo Fortino. Towards collaborative robotics in top view surveillance: A framework for multiple object tracking by detection using deep learning. *IEEE/CAA Journal of Automatica Sinica*, 8(7):1253–1270, 2021. 1

[3] Mk Bashar, Samia Islam, Kashifa Kawaakib Hussain, Md. Bakhtiar Hasan, A. B. M. Ashikur Rahman, and Md. Hasanul Kabir. Multiple object tracking in recent times: A literature review, 2022. 1

[4] Keni Bernardin and Rainer Stiefelhagen. Evaluating multiple object tracking performance: The clear mot metrics. *EURASIP Journal on Image and Video Processing*, 2008, 01 2008. 5

[5] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple online and realtime tracking. In *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE, Sept. 2016. 2

[6] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection, 2020. 2, 4

[7] Malik Boudiaf, Jérôme Rony, Imtiaz Masud Ziko, Eric Granger, Marco Pedersoli, Pablo Piantanida, and Ismail Ben Ayed. A unifying mutual information view of metric learning: cross-entropy vs. pairwise losses, 2021. 4

[8] Mohamed Chaabane, Peter Zhang, J. Ross Beveridge, and Stephen O'Hara. Deft: Detection embeddings for tracking, 2021. 1

[9] Gal Chechik, Varun Sharma, Uri Shalit, and Samy Bengio. Large scale online learning of image similarity through ranking. *Journal of Machine Learning Research*, 11(Mar):1109–1135, 2010. 2

[10] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations, 2020. 4

[11] Gioele Ciaparrone, Francisco Luque Sánchez, Siham Tabik, Luigi Troiano, Roberto Tagliaferri, and Francisco Herrera. Deep learning in video multi-object tracking: A survey. *Neurocomputing*, 381:61–88, Mar. 2020. 1

[12] Anthony Cioppa, Silvio Giancola, Adrien Deliege, Le Kang, Xin Zhou, Zhiyu Cheng, Bernard Ghanem, and Marc Van Droogenbroeck. Soccernet-tracking: Multiple object tracking dataset and benchmark in soccer videos, 2022. 1

[13] Patrick Dendorfer, Aljoša Ošep, Anton Milan, Konrad Schindler, Daniel Cremers, Ian Reid, Stefan Roth, and Laura Leal-Taixé. Motchallenge: A benchmark for single-camera multiple target tracking, 2020. 5

[14] Patrick Dendorfer, Hamid Rezatofighi, Anton Milan, Javen Shi, Daniel Cremers, Ian Reid, Stefan Roth, Konrad Schindler, and Laura Leal-Taixé. Mot20: A benchmark for multi object tracking in crowded scenes, 2020. 5

[15] Piotr Dollar, Christian Wojek, Bernt Schiele, and Pietro Perona. Pedestrian detection: A benchmark. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 304–311, 2009. 5

[16] Yunhao Du, Zhicheng Zhao, Yang Song, Yanyun Zhao, Fei Su, Tao Gong, and Hongying Meng. Strongsort: Make deepsort great again, 2023. 5

[17] Andreas Ess, Bastian Leibe, Konrad Schindler, and Luc Van Gool. A mobile vision system for robust multi-person tracking. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008. 5

[18] Tobias Fischer, Thomas E. Huang, Jiangmiao Pang, Linlu Qiu, Haofeng Chen, Trevor Darrell, and Fisher Yu. Qdtrack: Quasi-dense similarity learning for appearance-only multiple object tracking, 2023. 1, 3, 8

[19] Dengpan Fu, Dongdong Chen, Jianmin Bao, Hao Yang, Lu Yuan, Lei Zhang, Houqiang Li, and Dong Chen. Unsupervised pre-training for person re-identification, 2021. 4

[20] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. Yolox: Exceeding yolo series in 2021, 2021. 2

[21] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning, 2020. 4

[22] Lingxiao He, Xingyu Liao, Wu Liu, Xinchen Liu, Peng Cheng, and Tao Mei. Fastreid: A pytorch toolbox for general instance re-identification. *arXiv preprint arXiv:2006.02631*, 2020. 3, 4

[23] Alexander Hermans, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification, 2017. 4

[24] Hou-Ning Hu, Qi-Zhi Cai, Dequan Wang, Ji Lin, Min Sun, Philipp Krähenbühl, Trevor Darrell, and Fisher Yu. Joint monocular 3d vehicle detection and tracking, 2019. 1

[25] Glenn Jocher. Ultralytics yolov5, 2020. 3

[26] Glenn Jocher and Jing Qiu. Ultralytics yolo11, 2024. 2

[27] Glenn Jocher, Jing Qiu, and Ayush Chaurasia. Ultralytics YOLO, Jan. 2023. 2

[28] Hyeonchul Jung, Seokjun Kang, Takgen Kim, and HyeongKi Kim. Conftrack: Kalman filter-based multi-person tracking by utilizing confidence score of detection box. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 6583–6592, 2024. 1, 3

[29] Zoran Kalafatić, Tomislav Hrkać, and Karla Brkić. Multiple object tracking for football game analysis. In *2022 45th Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO)*, pages 936–941, 2022. 1

[30] R. E. Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1):35–45, 03 1960. 2

[31] H. W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2):83–97, 1955. 2

[32] Jiaxin Li, Yan Ding, and Hualiang Wei. Simpletrack: Rethinking and improving the jde approach for multi-object tracking, 2022. 1, 3, 8

[33] Chao Liang, Zhipeng Zhang, Xue Zhou, Bing Li, and Weiming Hu. One more check: Making "fake background" be tracked again, 2021. 1, 3, 8

[34] Chao Liang, Zhipeng Zhang, Xue Zhou, Bing Li, Shuyuan Zhu, and Weiming Hu. Rethinking the competition between detection and reid in multi-object tracking, 2022. 1, 3, 8

[35] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection, 2018. 3

[36] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2015. 5

[37] Yutian Lin, Lingxi Xie, Yu Wu, Chenggang Yan, and Qi Tian. Unsupervised person re-identification via softened similarity learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3387–3396, 2020. 4

[38] Zhichao Lu, Vivek Rathod, Ronny Votel, and Jonathan Huang. Retinatrack: Online single stage joint detection and tracking, 2020. 1, 3

[39] Jonathon Luiten, Aljosa Osep, Patrick Dendorfer, Philip Torr, Andreas Geiger, Laura Leal-Taixé, and Bastian Leibe. Hota: A higher order metric for evaluating multi-object tracking. *International Journal of Computer Vision*, 129(2):548–578, Oct. 2020. 5

[40] Anton Milan, Laura Leal-Taixe, Ian Reid, Stefan Roth, and Konrad Schindler. Mot16: A benchmark for multi-object tracking, 2016. 5

[41] Kevin Musgrave, Serge J. Belongie, and Ser-Nam Lim. Pytorch metric learning. *ArXiv*, abs/2008.09164, 2020. 5

[42] Ricardo Pereira, Guilherme Carvalho, Luís Garrote, and Urbano Nunes. Sort and deep-sort based multi-object tracking for mobile robotics: Evaluation with new data association metrics. *Applied Sciences*, 12:1319, 01 2022. 1

[43] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement, 2018. 3

[44] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks, 2016. 2

[45] Weihong Ren, Denglu Wu, Hui Cao, Xi'ai Chen, Zhi Han, and Honghai Liu. Joint counting, detection and re-identification for multi-object tracking, 2024. 1, 3, 8

[46] Ergys Ristani, Francesco Solera, Roger S. Zou, Rita Cucchiara, and Carlo Tomasi. Performance measures and a data set for multi-target, multi-camera tracking, 2016. 5

[47] Peter J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987. 5

[48] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015. 2

[49] Shuai Shao, Zijian Zhao, Boxun Li, Tete Xiao, Gang Yu, Xiangyu Zhang, and Jian Sun. Crowdhuman: A benchmark for detecting human in a crowd, 2018. 5

[50] Vukasin D Stanojevic and Branimir T Todorovic. Boosttrack: boosting the similarity measure and detection confidence for improved multiple object tracking. *Machine Vision and Applications*, 35(3), 2024. 1, 3, 5

[51] Oliver Urbann, Oliver Bredtmann, Maximilian Otten, Jan-Philip Richter, Thilo Bauer, and David Zibriczky. Online and real-time tracking in a surveillance scenario, 2021. 1

[52] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding, 2019. 3, 4

[53] Kanav Vats, Pascale Walters, Mehrnaz Fani, David A. Clausi, and John Zelek. Player tracking and identification in ice hockey, 2021. 1

[54] Jian Wang, Feng Zhou, Shilei Wen, Xiao Liu, and Yuanqing Lin. Deep metric learning with angular loss, 2017. 4

[55] Xun Wang, Xintong Han, Weilin Huang, Dengke Dong, and Matthew R. Scott. Multi-similarity loss with general pair weighting for deep metric learning, 2020. 4

[56] Yu-Hsiang Wang, Jun-Wei Hsieh, Ping-Yang Chen, Ming-Ching Chang, Hung Hin So, and Xin Li. Smiletrack: Similarity learning for occlusion-aware multiple object tracking, 2024. 1, 3

[57] Zhongdao Wang, Liang Zheng, Yixuan Liu, Yali Li, and Shengjin Wang. Towards real-time multi-object tracking, 2020. 1, 3, 5, 8

[58] Justin Wilson and Ming C. Lin. Avot: Audio-visual object tracking of multiple objects for robotics. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10045–10051, 2020. 1

[59] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric, 2017. 3

[60] Tong Xiao, Shuang Li, Bochao Wang, Liang Lin, and Xiaogang Wang. Joint detection and identification feature learning for person search, 2017. 5

[61] Fan Yang, Shigeyuki Odashima, Shoichi Masui, and Shan Jiang. Hard to track objects with irregular motions and similar appearances? make it easier by buffering the matching space, 2023. 3

[62] Sisi You, Hantao Yao, Bing-kun Bao, and Changsheng Xu. Utm: A unified multiple object tracking model with identity-aware feature enhancement. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21876–21886, 2023. 3

[63] En Yu, Zhuoling Li, Shoudong Han, and Hongwei Wang. Relationtrack: Relation-aware multiple object tracking with decoupled representation, 2021. 1, 3, 8

[64] Fengwei Yu, Wenbo Li, Quanquan Li, Yu Liu, Xiaohua Shi, and Junjie Yan. Poi: Multiple object tracking with high performance detection and appearance feature, 2016. 3

[65] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization, 2018. 3

[66] Shanshan Zhang, Rodrigo Benenson, and Bernt Schiele. Citypersons: A diverse dataset for pedestrian detection, 2017. 5

[67] Yunfei Zhang, Chao Liang, Jin Gao, Zhipeng Zhang, Weiming Hu, Stephen Maybank, Xue Zhou, and Liang Li. Temporal correlation meets embedding: Towards a 2nd generation of jde-based real-time multi-object tracking, 2024. 1, 3, 8

[68] Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Fucheng Weng, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. Bytetrack: Multi-object tracking by associating every detection box, 2022. 2

[69] Yifu Zhang, Chunyu Wang, Xinggang Wang, Wenjun Zeng, and Wenyu Liu. Fairmot: On the fairness of detection and re-identification in multiple object tracking. *International Journal of Computer Vision*, 129(11):3069–3087, Sept. 2021. 1, 3, 8

[70] Liang Zheng, Hengheng Zhang, Shaoyan Sun, Manmohan Chandraker, Yi Yang, and Qi Tian. Person re-identification in the wild, 2017. 5

[71] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points, 2019. 2