Human Behavior, Development and Society
E-ISSN 2651-1762, Vol 25 No 3, December 2024

Enhancing Multi-Object Tracking with Compact Model Adjustment

Pimpa Cheewaprakobkit, Asia-Pacific International University, Thailand

Date Received: 5 September 2024 Revised: 11 November 2024 Accepted: 18 November 2024

Abstract

Tracking human movement and interactions in complex environments is a key challenge in computer vision, especially for multi-object tracking. Transformer-based models have shown promise in addressing these challenges due to their capacity to recognize complex patterns across sequences. However, their high computational demands and substantial training data requirements often restrict their real-world applicability. This study aimed to enhance multi-object tracking by introducing a Compact Model Adjustment approach that integrates trainable rank-decomposition matrices within the Transformer architecture. This approach involves freezing the pre-trained model weights and adding trainable low-rank matrices to each layer, substantially reducing the number of parameters that need updating during training. This design allows the model to retain its pre-trained knowledge while efficiently adapting to new tasks, thereby reducing the overall computational load. Additionally, the proposed approach utilizes data from both the current and previous frames to refine object localization and association. Experimental results on the MOT17 benchmark demonstrated that this method achieved a Multiple Object Tracking Accuracy of 71.0, comparable to state-of-the-art techniques while enhancing computational efficiency. This work provides a practical solution for real-world applications in areas such as surveillance, autonomous driving, and sports analytics.

Keywords: *Multi-object tracking, compact model adjustment, transformer architecture*

Introduction

Multi-object tracking (MOT) (Amosa et al., 2023) has become an essential area of research in computer vision, driven by the increasing demand for automated systems capable of monitoring and analyzing dynamic environments. The ability to accurately track multiple objects over time is crucial for a wide range of applications, such as surveillance, robotics, autonomous driving, sports analytics, and human-computer interaction. The primary challenge in MOT lies in accurately detecting and tracking multiple objects across frames in a video sequence, especially in dynamic environments where occlusions, varying object appearances, and complex interactions occur (Du et al., 2024). Recent advancements in deep learning, particularly through the use of transformer architectures, have shown promise in addressing these challenges by effectively capturing temporal dependencies and spatial relationships among objects (Manakitsa et al., 2024).

The introduction of transformer models has revolutionized various domains, including natural language processing and computer vision, by utilizing self-attention mechanisms to capture long-range dependencies and contextual information (Patwardhan et al., 2023). One of the notable contributions in this domain is the DEtection TRansformer (DETR), which models object detection as a translation task, converting image features into object-level representations (Carion et al., 2020). DETR's innovative approach allows it to handle complex scenes more effectively than traditional detection methods, laying the groundwork for subsequent models that extend its capabilities to multi-object tracking.

In addition to DETR, other transformer-based models have also made significant strides in MOT. For example, the TrackFormer model (Meinhardt et al., 2022) employs a transformer architecture to simultaneously perform object detection and tracking. It utilizes a bipartite matching algorithm to associate detected objects across frames, effectively leveraging the strengths of transformers to enhance tracking accuracy in dynamic scenarios. Similarly, the TransTrack model (Sun et al., 2020) employs a transformer-based architecture for simultaneous object detection and tracking. By

leveraging the self-attention mechanism, this model can capture relationships between objects across frames, enhancing tracking performance in complex scenarios.

Despite these advancements, the computational demands of transformer-based models have remained a concern, especially when they are deployed in real-time applications. The need for extensive training data and high computational resources can limit their practicality. To address these challenges, researchers have explored various adaptation techniques that reduce the number of trainable parameters while preserving model performance. For example, the Dynamic Layer Tying technique (Hay & Wolf, 2024) involves dynamically selecting layers during training and tying them together, facilitating weight sharing across layers. By employing reinforcement learning to determine whether to train each layer independently or to share weights from previous layers, this method can significantly reduce the number of trainable parameters. However, the reliance on reinforcement learning can introduce complexity in implementation and may require careful tuning to achieve optimal performance.

Another technique is Teacher-Student Learning (Wang et al., 2019), where a larger, more complex model (teacher) is trained first, and then its knowledge is transferred to a smaller, more efficient model (student). The student model can be trained to mimic the teacher's output, often using techniques like knowledge distillation. While these techniques can effectively reduce the number of trainable parameters in transformers, they also have potential disadvantages, such as performance degradation, where techniques like aggressive pruning or quantization (Wu et al., 2019) can lead to a decrease in model performance. In a related study, Hu et al. (2021) proposed a low-rank adaptation approach in natural language processing (NLP), which decomposes large weight matrices into smaller ones. This technique can significantly reduce the number of parameters and the computational complexity of the model.

To address these limitations, the present researcher proposed a novel approach termed Compact Model Adjustment (CMA), which integrates the concept of trainable rank-decomposition matrices into the transformer encoder architecture. The trainable rank-decomposition matrices technique allows for the efficient adaptation of large pre-trained models by freezing the original model weights and adding trainable rank-decomposition matrices to each layer of the transformer architecture. This method drastically reduces the number of trainable parameters, thereby minimizing computational overhead while maintaining or even improving model performance. The rationale behind CMA comes from the observation that not all parameters in a model contribute equally to its performance on specific tasks. By focusing on updates to trainable rank-decomposition matrices, CMA enables models to adapt effectively to new tasks without the need for extensive retraining of all parameters. This is particularly beneficial for multi-object tracking, where rapid adaptation to varying conditions and object behaviors is essential.

The key contributions can be outlined as follows:

- An architectural innovation was proposed that incorporated the Compact Model Adjustment technique specifically designed for multi-object tracking (MOT). This technique addresses the challenges of high computational complexity and extensive training data requirements commonly associated with transformer-based models.
- A pair of consecutive video frames was utilized, specifically the current frame and the previous frame, to enhance object tracking and association.
- The performance of the proposed method was evaluated by comparing it to state-of-the-art techniques on the MOT17 benchmark, demonstrating the effectiveness of the approach in real-world tracking scenarios.

The structure of the remaining manuscript is outlined as follows: Section 2 introduces the related work, Section 3 presents the proposed method, Section 4 demonstrates the experimental results, and Section 5 concludes this work.

Related Work

Multi-Object Tracking in Complex Environments

Multi-Object Tracking has seen significant advancements over the years, particularly in dealing with complex environments. These environments, characterized by occlusions, varying object appearances, and dynamic backgrounds, pose substantial challenges to tracking algorithms. One of the foundational approaches to MOT has been the Kalman Filter (Khodarahmi & Maihami, 2023), which provides a recursive solution for estimating the state of a dynamic system in the presence of noise. Although it was primarily developed for single-object tracking, it has been adapted for multiple objects by employing data association techniques (Yilmaz et al., 2006). In recent years, deep learning has revolutionized MOT, especially with the introduction of deep appearance models that enhance the robustness of data association (Emami et al., 2020).These models can learn discriminative features from object appearances, which are crucial in distinguishing between similar objects in crowded scenes. Moreover, approaches like the SORT (Simple Online and Realtime Tracking) algorithm (Bewley et al., 2016), have demonstrated the effectiveness of combining a Kalman Filter with a Hungarian algorithm for data association in real-time tracking scenarios. This method has been further improved by extending it with deep learning-based re-identification models, resulting in DeepSORT (Wojke et al., 2017).

Transformer-Based Models in Computer Vision

Transformers, initially developed for natural language processing, have recently gained popularity in computer vision tasks due to their capacity to capture long-range dependencies and contextual information across sequences (Meinhardt et al., 2022). In the context of MOT, transformer-based models offer a promising solution by utilizing attention mechanisms to effectively associate objects across frames, even in complex and crowded scenes. These models, however, come with their own set of challenges. The computational complexity of transformers is quadratic with respect to input size, making them resource-intensive and requiring extensive training data to achieve optimal performance (Fournier et al., 2023) .

Challenges and Limitations of Transformer Models in MOT

Despite their potential, transformer-based models for MOT face significant hurdles in real-world applications. One major limitation is the high computational cost associated with training and deploying these models. As the input sequence length increases, the memory and processing power required grows substantially, often making real-time applications impractical (Carion et al., 2020). Moreover, transformers require large amounts of labeled data to avoid overfitting, which can be a bottleneck in domains where annotated data is scarce or expensive to obtain (Alzubaidi et al., 2023).

Advancements in Reducing Computational Complexity

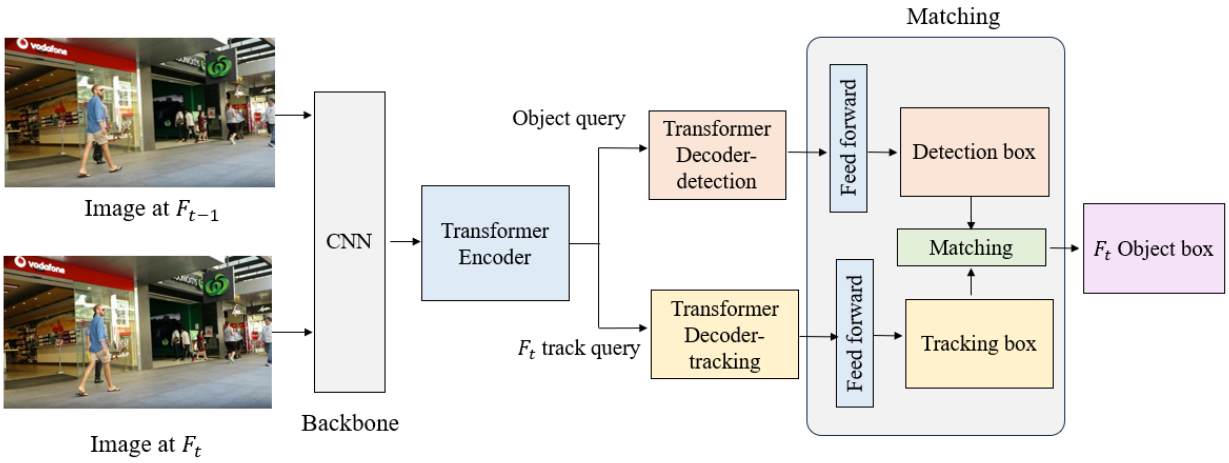
Recent advancements in MOT have focused on reducing computational complexity to enable real-time applications. One approach has involved model pruning, where redundant parameters are removed from deep neural networks (Wu et al., 2019). Another technique is quantization, which reduces the precision of model weights and activations to smaller data types, resulting in smaller models and faster inference (Xie et al., 2023). While parameter pruning and quantization are effective techniques for reducing the size and computational complexity of deep learning enhancement models, they can also lead to performance degradation, require careful hyperparameter tuning, and may not be fully compatible with all hardware platforms or generalize well to different datasets or noise conditions. Additionally, knowledge distillation transfers knowledge from a large, complex model to a smaller, more efficient one (Faber et al., 2024). These techniques have shown promising results in reducing computational complexity. However, they may require fine-tuning and careful design to achieve optimal performance, which can be time-consuming and challenging.

Proposed Method

The proposed method was based on encoder-decoder transformer architecture. It leverages the power of transformers to simultaneously perform detection and tracking within a unified framework. By treating object tracking as a sequence prediction problem, the researcher utilized transformers to model the relationships between detected objects across consecutive frames. The encoder-decoder transformer architecture consisted of multi-head attention layers, residual connections, layer normalization, feed-forward networks, and cross-attention. Multi-head attention, a mechanism for self-attention, operates in parallel multiple times. The overall architecture of the proposed method is illustrated in Figure 1.

The process begins with a pair of consecutive video frames, typically denoted as the current frame F_t and the previous frame F_{t-1}. Both frames are processed by a convolutional neural network (CNN) backbone network, specifically employing ResNet50, to extract deep features. These features capture various aspects of the objects in the scene, such as their appearance, shape, and position.

Figure 1 *Overall Architecture of the Proposed Method*



Transformer Encoder

The extracted features from both frames are flattened and combined, then passed through a transformer encoder. The encoder processes these features to capture global contextual information and interactions between objects within and across the frames. The encoder's role is to build rich feature representations that are aware of the spatial and temporal relationships in the input frames.

The researcher introduced Compact Model Adjustment, a method that employs trainable rank-decomposition matrices to significantly reduce the number of trainable parameters for downstream tasks when applied to multi-head attention mechanisms in transformer encoders.

Compact Model Adjustment

The researcher introduced a CMA method that was designed to efficiently adapt large models to new tasks or domains without requiring full fine-tuning. Full fine-tuning can be computationally expensive, time-consuming, and storage-intensive. The CMA method chosen addressed these issues by incorporating trainable rank-decomposition matrices into the pre-trained model's parameters while freezing the original weights. This approach allows for efficient adaptation while preserving the valuable knowledge stored in the pre-trained model.

Applying CMA to a transformer's multi-head attention mechanism involves a series of steps to adapt the pre-trained model weights with updates to trainable rank-decomposition matrices. In transformers, the multi-head attention mechanism is a crucial component. It consists of several attention heads that allow the model to focus on different parts of the input sequence. Each head computes the following (1):

$$Attention\ (Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (1)$$

Where $Q=XW_Q$, $K=XW_K$, $V=XW_V$, X is the input, W_Q, W_K and W_K are learned weight matrices for Queries, Keys, and Values, respectively, and d_k is the dimensionality of the Key Vectors.

In CMA, the weight matrix W (for any of W_Q , W_K , W_V) is decomposed into two parts as shown in (2).

$$W'=W+\Delta W \quad (2)$$

Where ΔW is an update of trainable rank-decomposition matrices represented as shown in (3).

$$\Delta W=B\times A \quad (3)$$

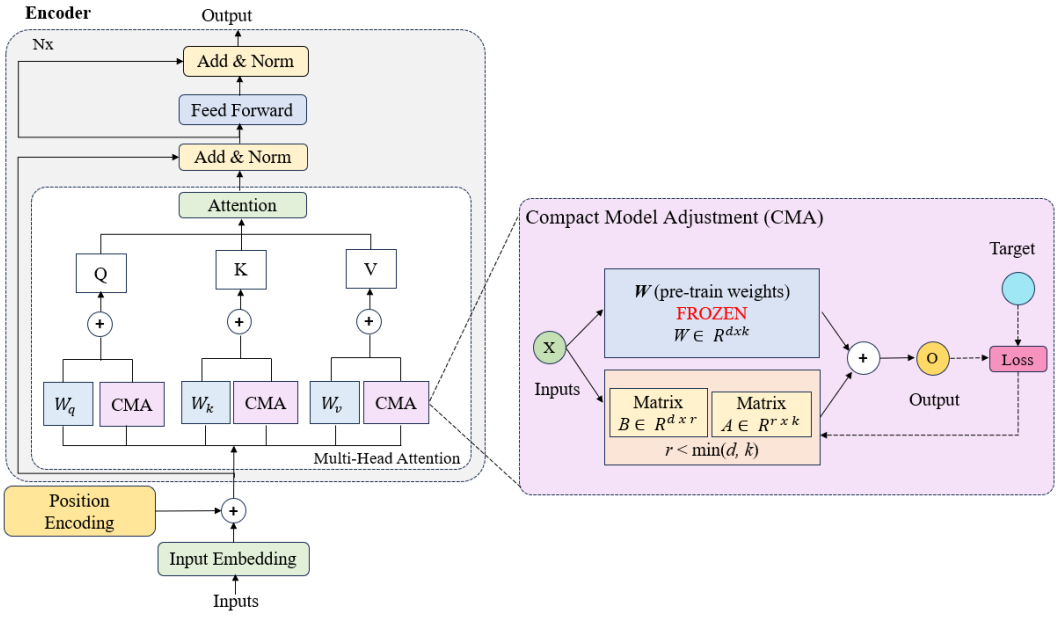
Where $B\in R^{d\times r}$ and $A\in R^{r\times d}$ are the trainable rank-decomposition matrices with *r* (the rank) being much smaller than the original dimension d. The idea is to train these trainable rank-decomposition matrices *A* and *B* while keeping the original *W* frozen.

The process begins with a pre-trained transformer model with weights *W* and initializing these trainable rank-decomposition matrices *B* and *A* for each attention head. During the forward pass, the input *X* is projected using the adapted weight matrices *W'* to compute the Queries, Keys, and Values as shown in (4).

$$Q' = X(W_Q+\Delta W_Q) = X(W_Q+B_QA_Q) \quad (4)$$

The same calculations apply for K' and V'. During training, only the trainable rank-decomposition matrices *B* and *A* are modified. The original weights W_Q, W_K, and W_V remain unchanged. By restricting the adaptation to the update of trainable rank-decomposition matrices, the number of trainable parameters is substantially decreased, resulting in more efficient fine-tuning while still enabling effective model adaptation. Figure 2 visually illustrates the Cross-Modal Attention mechanism within the encoder transformer's multi-head attention. CMA utilizes a bounding box regression loss function to minimize the error between the predicted and ground truth bounding boxes.

Figure 2 *Illustration of the Compact Model Adjustment in the Encoder Transformer*

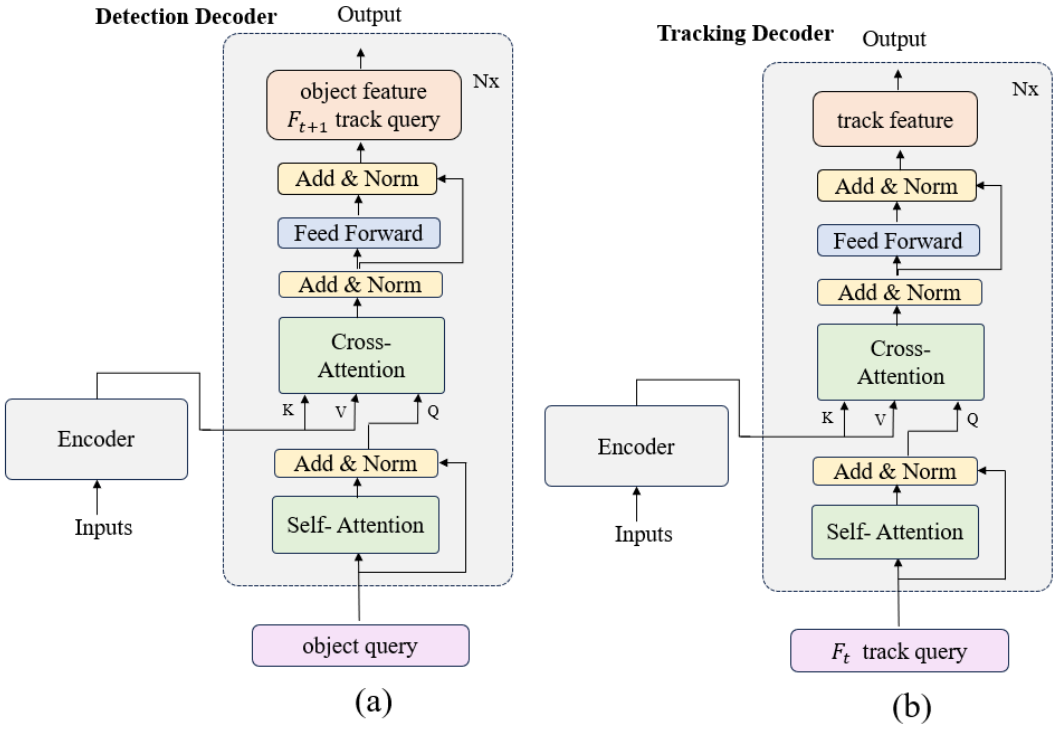


Detection Decoder

The detection decoder is initialized with a set of object queries, which are learnable embeddings that the model uses to search for objects within the encoded features. These queries aid in identifying and locating objects in the current frame. The detection decoder processes the encoded features and queries to generate object predictions. Specifically, for each query, the decoder outputs a set of

bounding boxes and class scores, representing the locations and categories of detected objects in the current frame. The detection decoder block's output comprises a set of bounding boxes indicating the objects' locations in the current frame, along with corresponding confidence scores for each detection, as illustrated in Figure 3(a).

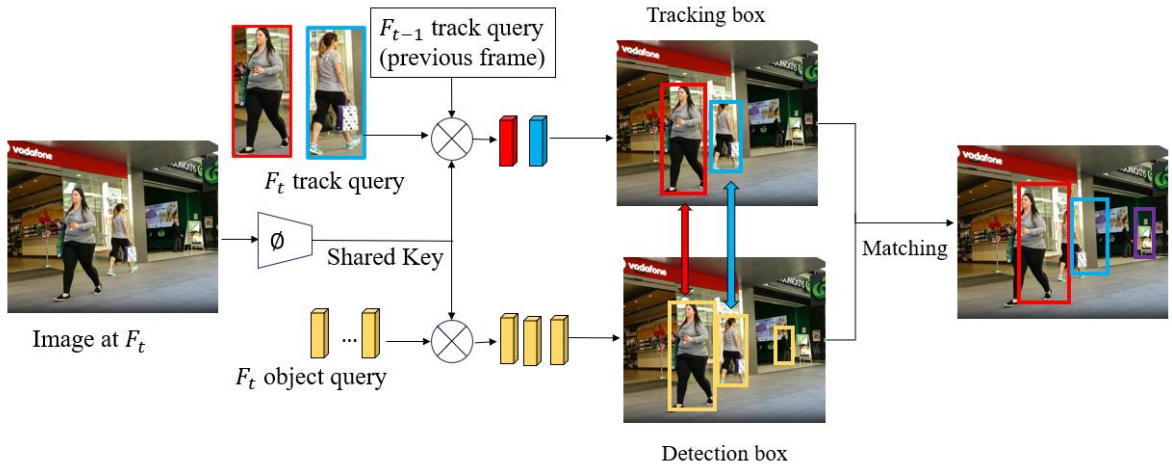
Figure 3 *Detection Decoder and Tracking Decoder Architecture*



Tracking Decoder

The tracking process utilizes track queries, which are derived from the detected objects in the previous frame. These track queries preserve the appearance and positional information of objects detected in F_{t-1}. The tracking decoder as shown in Figure 3(b), identical in architecture to the detection decoder as shown in Figure 3(a), employs these track queries to associate objects detected in the previous frame with those in the current frame. It updates the positions of these objects and generates tracking boxes for each object in F_t. The output of the tracking decoder consists of a set of bounding boxes that indicate the tracked objects' positions in the current frame, linking them to the detections from the previous frame as illustrated in Figure 4.

Figure 4 *Overview of Object Detection and Object Tracking*



In Figure 4, the image feature maps act as a shared key. This key represents specific regions within the image that correspond to the features of an object. These features are crucial for determining the relevance of this object to other objects within a sequence. The "object query" refers to the representation of the object that the model is focusing on or interested in at a particular moment. The attention mechanism uses these keys (feature maps) and queries (object representations) to determine which parts of the image are most relevant to the task at hand.

Matching

The matching model associates detected objects in the current frame with tracked objects from the previous frame. This association is based on the bounding boxes and the feature similarities between the frames. The final output consists of the detected and tracked objects, where each tracked object is linked to its corresponding detection in the current frame. The model outputs the updated trajectories for all objects, which are continuously refined as the video progresses. To match detection boxes and tracking boxes, the Hungarian algorithm is used based on Intersection over Union (IoU) similarity.

Training Loss

Loss Function

The training of the model involves a combination of loss functions to train the model. The loss function consists of two main components: a classification loss for detecting the presence of objects and a bounding box regression loss for accurately predicting the locations of the objects.

Classification Loss

Classification loss measures the model's ability to correctly identify objects. Equation (5) defined the calculation of classification loss (Singh & Príncipe, 2010).

$$L_{cls} = -\alpha_t (1 - p_t)^{\gamma} \log(p_t) \quad (5)$$

In this equation, p_t represents the model's predicted probability for the correct class. The parameter α_t serves as a weighting factor to balance the significance of positive and negative examples. Meanwhile, γ is a focusing parameter that reduces the loss for well-classified examples, making the model focus on challenging cases.

Bounding Box Regression Loss

For bounding box regression, the researcher used a combination of L1 loss and Generalized Intersection over Union (GIoU) loss to evaluate the difference between the predicted bounding boxes

and the ground truth boxes. The L1 loss specifically quantifies the absolute difference between the predicted and actual bounding box coordinates. It is defined as the average absolute error across all bounding boxes, where the loss is computed as the sum of the absolute differences between each predicted bounding box and its corresponding ground truth, divided by the total number of bounding boxes. The L1 loss for bounding box regression (Zhai et al., 2020) is defined in equation (6):

$$L_{L1} = \frac{1}{N} \sum_{i=1}^{N} |y_i - \hat{y}_i| \quad (6)$$

In this context, y_i represents the ground truth bounding box, $\hat{y}_i$ denotes the predicted bounding box, and *N* is the number of bounding boxes.

Generalized Intersection over Union (GIoU) Loss

The GIoU loss (Cao, 2021) is defined in equation (7):

$$L_{GIoU} = 1 - \frac{|Intersection|}{|Union|} + \frac{|C-U|}{|C|} \quad (7)$$

In this formula, |Intersection |*Intersection*| represents the area of overlap between the predicted and ground truth bounding boxes, while |*Union*| denotes the total area covered by both boxes combined. The term |*C*| refers to the area of the smallest enclosing box that contains both the predicted and ground truth boxes, and |*U*| corresponds to the area of the union of the two boxes. The GIoU loss incorporates these elements to provide a more comprehensive measure of the alignment between the predicted and ground truth bounding boxes.

Overall Loss Function

The overall loss function used for training is a combination of several components, as represented in equation (8).

$$L = L_{cls} + L_{l1} + L_{GIoU} \quad (8)$$

In this formulation, L_{cls} referred to the classification loss, L_{l1} denoted the loss associated with bounding box regression. Lastly, GIoU represented the Generalized Intersection over Union loss. Together, these losses guide the training process to optimize both object classification and bounding box accuracy.

Experimental results

Datasets

The MOT17 dataset is a widely recognized benchmark for evaluating the performance of multi-object tracking (MOT) algorithms. As a publicly available dataset, MOT17 serves as a valuable resource for researchers to develop and improve their algorithms. It comprises a collection of video sequences with annotated bounding boxes, where the primary objective is to track multiple objects, typically pedestrians, across frames. Experiments were conducted on the pedestrian-tracking dataset MOT17. The researcher utilized the standard split of the MOT17 dataset, which included a training set and a validation set. The MOT17 dataset consisted of seven training sequences and seven test sequences, each representing a distinct video clip featuring people navigating various environments, such as streets, squares, or other public spaces. The dataset was specifically designed to offer diverse and challenging scenarios for testing multi-object tracking algorithms. The training sequences were provided with ground truth annotations, indicating the trajectories of objects, while the test sequences were supplied without ground truth annotations. The evaluation of the test set was performed by submitting the results to the MOT Challenge website. The MOT17 test set included 2,355 trajectories spread across 17,757 frames. Trajectories represented the paths or sequences of positions that objects (such as people) follow over time as they move through different frames in a video.

Evaluation Metrics

To evaluate the performance of this proposed method, the researcher utilized the widely recognized MOT metrics set for quantitative assessment. The primary metric is Multiple-Object Tracking Accuracy (MOTA), which measures overall tracking performance. Identity Switches (ID Sw) quantify the number of instances where an object's identity is mistakenly assigned to another object. This issue arises when the tracker loses track of an object and subsequently re-detects it, leading to the assignment of a different ID. The Identity F1 Score (IDF1) represents the harmonic mean of precision and recall for object identity, offering a balanced evaluation of how effectively the tracker assigns identities to objects. Multiple Object Loss (ML) measures the frequency with which an object is incorrectly associated with another object or fails to be associated with any object at all; a lower ML value signifies better tracking accuracy. Finally, Missed Targets (MT) counts the instances where an object goes undetected by the tracker, with a lower MT value indicating superior object detection accuracy.

Implementation Details

The researcher utilized the ResNet-50 architecture as the backbone of the network. ResNet-50 is a widely-used convolutional neural network known for its depth and ability to learn complex patterns, which enhances model robustness and generalization. To further improve generalization and prevent overfitting, data augmentation techniques were incorporated such as random horizontal flipping, random cropping, scaling, and resizing of input images. These techniques artificially expand the size and diversity of the training dataset by applying random transformations to the images. The AdamW optimizer, initialized with a learning rate of $2.0\ x \cdot 10^{-4}$, was employed to guide the model's learning process. The model was trained for 100 epochs.

Comparison

The performance comparison between the proposed method and other state-of-the-art methods on the MOT17 dataset is summarized in Table I. The proposed method achieved a Multiple-Object Tracking Accuracy (MOTA) of 71.0, surpassing all other methods listed, indicating a superior overall tracking performance. While this study's Identity F1 Score (IDF1) of 63.8 is slightly lower than SUSHI (71.5), it still reflected a robust balance between precision and recall in identity assignment.

In terms of Missed Targets (MT), this method successfully tracked 1,038 targets, which is among the highest, second only to NCT, which tracked 1,092 targets. This demonstrated the method's strong ability to maintain object tracking across sequences. Additionally, this method had a relatively low number of Multiple Object Losses (ML) at 464, which is among the lowest, although NCT achieved an even lower ML of 399. This further highlighted the method's tracking consistency.

Regarding Identity Switches (ID Sw.), this study's method recorded 4,113 switches, which, although not the lowest, showed a reasonable performance when considering the complexity of maintaining consistent object identities. Overall, this study's method demonstrated a leading performance in MOTA and a strong balance across other key metrics, making it highly effective in addressing the challenges posed by the MOT17 dataset.

Table 1 *Comparison of the Proposed Method and Other Methods Evaluated on MOT17 (Arrows Indicating Low or High Optimal Metric Values)*

Method	MOTA ↑	IDF1 ↑	MT ↑	ML ↓	ID Sw. ↓
TADN (Psalta et al., 2024)	54.6	49.0	528	711	4869
IQHAT (He et al., 2022)	58.4	61.8	568	829	1261
UnsupTrack (Karthik et al., 2020)	61.7	58.1	640	760	1864
SUSHI (Karthik et al., 2020)	62.0	71.5	801	741	1041
MPTC (Stadler & Beyerer, 2021)	62.6	65.8	627	750	4074
UTM (You et al., 2023)	63.5	65.1	881	635	1686
Hugmot (Wan et al., 2021)	64.8	62.8	738	645	2102
BYTE_Pub (Wan et al., 2021)	67.4	70.0	730	735	1331
OUTrack_fm_p (Liu et al., 2022)	69.0	66.8	885	464	4472
NCT (Zeng et al., 2023)	69.5	68.5	1092	399	4919
PixelGuided (Boragule et al., 2022)	69.7	68.4	903	615	3639
Proposed method	**71.0**	**63.8**	**1038**	**464**	4113

Note. The range for Multiple-Object Tracking Accuracy (MOTA) scores is 0–100; model accuracy is being reported.

Conclusion

In this study, the challenge of Multi-Object Tracking in complex environments was addressed by proposing a Compact Model Adjustment approach that enhanced the efficiency of transformer-based models. The results demonstrated that incorporating trainable rank-decomposition matrices into transformer architecture is an effective strategy for balancing model complexity and performance. By leveraging data from both the current and previous frames, this study's approach improved the model's ability to track objects over time, thereby addressing some of the critical challenges associated with MOT in dynamic and complex environments. The performance of the model was comparable to state-of-the-art techniques, highlighting the potential of this approach for real-world applications where computational resources may be limited.

Despite the promising results, this study's approach has limitations. This method is applicable only after the full model has been trained, limiting its use in scenarios where training from scratch is required. Future work should focus on extending the CMA approach to incorporate multi-camera setups or integrating 3D tracking capabilities to enhance the model's robustness in scenarios with significant occlusions or complex interactions between tracked objects.

References

Alzubaidi, L., Bai, J., Al-Sabaawi, A., Santamaría, J., Albahri, A. S., Al-dabbagh, B. S. N., Fadhel, M. A., Manoufali, M., Zhang, J., Al-Timemy, A. H., Duan, Y., Abdullah, A., Farhan, L., Lu, Y., Gupta, A., Albu, F., Abbosh, A., & Gu, Y. (2023). A survey on deep learning tools dealing with data scarcity: Definitions, challenges, solutions, tips, and applications. *Journal of Big Data*, *10*(1), 1–82. https://doi.org/10.1186/s40537-023-00727-2

Amosa, T. I., Sebastian, P., Izhar, L. I., Ibrahim, O., Ayinla, L. S., Bahashwan, A. A., Bala, A., & Samaila, Y. A. (2023). Multi-camera multi-object tracking: A review of current trends and future advances. *Neurocomputing*, *552*, 126558. https://doi.org/https://doi.org/10.1016/j.neucom.2023.126558

Bewley, A., Ge, Z., Ott, L., Ramos, F., & Upcroft, B. (2016, September 25–28). *Simple online and realtime tracking* [Paper presentation]. 2016 IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA. https://doi.10.1109/ICIP.2016.7533003

Boragule, A., Jang, H., Ha, N., & Jeon, M. (2022). Pixel-guided association for multi-object tracking. *Sensors*, *22*(22), 8922. https://www.mdpi.com/1424-8220/22/22/8922

Cao, Q. (2021, July 28–30). *Experimental study on the effect of loss function on object detection* [Paper presentation]. Proceedings of the 2021 International Conference on Pattern Recognition and Intelligent Systems, Bangkok, Thailand. https://doi.org/10.1145/3480651.3480690

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., & Zagoruyko, S. (2020, August 23–28). *End-to-End Object Detection with Transformers* [Paper presentation]. Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, Proceedings, Part I, Glasgow, United Kingdom. https://doi.org/10.1007/978-3-030-58452-8_13

Du, C., Lin, C., Jin, R., Chai, B., Yao, Y., & Su, S. (2024). Exploring the state-of-the-art in multi-object tracking: A comprehensive survey, evaluation, challenges, and future directions. *Multimedia Tools and Applications, 83,* 73151–73189. https://doi.org/10.1007/s11042-023-17983-2

Emami, P., Pardalos, P. M., Elefteriadou, L., & Ranka, S. (2020). Machine learning methods for data association in multi-object tracking. *ACM Computing Survey*, *53*(4), 1–34. https://doi.org/10.1145/3394659

Faber, N. G., Ziabari, S. S. M., & Nejadasl, F. K. (2024). Leveraging foundation models via knowledge distillation in multi-object tracking: Distilling DINOv2 features to FairMOT. *ArXiv*, *abs/2407.18288*. https://doi:10.48550/arXiv.2407.18288

Fournier, Q., Caron, G. M., & Aloise, D. (2023). A practical survey on faster and lighter transformers. *ACM Computing Survey*, *55*(14s), 1–40. https://doi.org/10.1145/3586074

Hay, T. D., & Wolf, L. (2024). Dynamic layer tying for parameter-efficient transformers. *ArXiv*, *abs/2401.12819*. https://arxiv.org/html/2401.12819v1

He, Y., Wei, X., Hong, X., Ke, W., & Gong, Y. (2022). Identity-quantity harmonic multi-object tracking. *IEEE Transactions on Image Processing*, *31*, 2201–2215. https://doi.org/10.1109/TIP.2022.3154286

Hu, J. E., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., & Chen, W. (2021). LoRA: Low-rank adaptation of large language models. *ArXiv*, *abs/2106.09685*. https://doi.org/10.48550/arXiv.2106.09685

Karthik, S., Prabhu, A., & Gandhi, V. (2020). Simple unsupervised multi-object tracking. *ArXiv*, *abs/2006.02609*. https://doi:10.48550/arXiv.2006.02609

Khodarahmi, M., & Maihami, V. (2023). A review on Kalman Filter Models. *Archives of Computational Methods in Engineering*, *30*(1), 727–747. https://doi.org/10.1007/s11831-022-09815-7

Liu, Q., Chen, D., Chu, Q., Yuan, L., Liu, B., Zhang, L., & Yu, N. (2022). Online multi-object tracking with unsupervised re-identification learning and occlusion estimation. *Neurocomput.*, *483*(C), 333–347. https://doi.org/10.1016/j.neucom.2022.01.008

Manakitsa, N., Maraslidis, G. S., Moysis, L., & Fragulis, G. F. (2024). A review of machine learning and deep learning for object detection, semantic segmentation, and human action recognition in machine and robotic vision. *Technologies*, *12*(2), 15. https://www.mdpi.com/2227-7080/12/2/15

Meinhardt, T., Kirillov, A., Leal-Taixé, L., & Feichtenhofer, C. (2022, June 18–24). *TrackFormer: Multi-object tracking with transformers.* 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) [pp. 8834–8844], New Orleans, Louisiana, USA. https://doi: 10.1109/CVPR52688.2022.00864

Patwardhan, N., Marrone, S., & Sansone, C. (2023). Transformers in the real world: A survey on NLP applications. *Information*, *14*(4), 242. https://doi.org/10.3390/info14040242

Psalta, A., Tsironis, V., & Karantzalos, K. (2024). Transformer-based assignment decision network for multiple object tracking. *Computer Vision and Image Understanding*, *241*(C), 103957. https://doi.org/10.1016/j.cviu.2024.103957

Singh, A., & Príncipe, J. C. (2010, July 18–23). *A loss function for classification based on a robust similarity metric.* The 2010 International Joint Conference on Neural Networks (IJCNN), Barcelona, Spain. https://doi:10.1109/IJCNN.2010.5596485

Stadler, D., & Beyerer, J. (2021, November 16–19). *Multi-pedestrian tracking with clusters*. 2021 17th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS) [pp. 1–10], Washington, District of Columbia, USA. https://doi: 10.1109/AVSS52988.2021.9663829.

Sun, P., Jiang, Y., Zhang, R., Xie, E., Cao, J., Hu, X., Kong, T., Yuan, Z., Wang, C., & Luo, P. (2020). TransTrack: Multiple-object tracking with transformer. *ArXiv*, *abs/2012.15460*. https://doi:10.48550/arXiv.2012.15460

Wan, X., Zhou, S., Wang, J., & Meng, R. (2021). *Multiple object tracking by trajectory map regression with temporal priors embedding.* Proceedings of the 29th ACM International Conference on Multimedia, Virtual Event [pp. 1377–1386], China. https://doi.org/10.1145/3474085.3475304

Wang, X., Hu, J., Lai, J., Zhang, J., & Zheng, W. (2019). Progressive teacher-student learning for early action prediction. Conference on Computer Vision and Pattern Recognition 2019 [pp. 3551–3560], Long Beach, California, United States. http://cvpr2019.thecvf.com/

Wojke, N., Bewley, A., & Paulus, D. (2017, September 17–20). *Simple online and realtime tracking with a deep association metric*. 2017 IEEE International Conference on Image Processing (ICIP) [pp. 3645–3649], Beijing, China. https://doi:10.1109/ICIP.2017.8296962

Wu, J. Y., Yu, C., Fu, S. W., Liu, C. T., Chien, S. Y., & Tsao, Y. (2019). Increasing compactness of deep learning based speech enhancement models with parameter pruning and quantization techniques. *IEEE Signal Processing Letters*, *26*(12), 1887–1891. https://doi.org/10.1109/LSP.2019.2951950

Xie, Y., Guo, Y., Hou, X., & Zheng, J. (2023, August 5–6). *Mixed-precision collaborative quantization for fast object tracking.* International Conference on Advances in Brain Inspired Cognitive Systems [pp. 229–238], Kuala Lumpur, Malaysia. https://doi.org/10.1007/978-981-97-1417-9_22

Yilmaz, A., Javed, O., & Shah, M. (2006). Object tracking: A survey. *ACM Computing Surveys, 38*(4), 13–es. https://doi.org/10.1145/1177352.1177355

You, S., Yao, H., Bao, B. k., & Xu, C. (2023, 17-24 June 2023). *UTM: A unified multiple object tracking model with identity-aware feature enhancement.* Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), (pp. 21876–21886). https://doi:10.1109/CVPR52729.2023.02095

Zeng, K., You, Y., Shen, T., Wang, Q., Tao, Z., Wang, Z., & Liu, Q. (2023). NCT: Noise-control multi-object tracking. *Complex & Intelligent Systems*, *9*(4), 4331–4347. https://doi.org/10.1007/s40747-022-00946-9

Zhai, H., Cheng, J., & Wang, M. (2020, December 11–13). *Rethink the IoU-based loss functions for bounding box regression.* 2020 IEEE 9th Joint International Information Technology and Artificial Intelligence Conference (ITAIC) [pp. 1522–1528], Chongqing, China. https://doi: 10.1109/ITAIC49862.2020.9339070