

# Synergizing RNN and Transformers with Exponential Decay Learning Rate for Accurate Multi-Object Tracking in Complex Environments

Dr. G. S. Gowri<sup>1</sup>, Dr. N. Mala<sup>2</sup>

<sup>1</sup>Associate Professor, Department of Information Technology, Kovai Kalaimagal College of Arts and Science, India, [gowrijanarthanan@gmail.com](mailto:gowrijanarthanan@gmail.com)

<sup>2</sup>Professor, Department of Mathematics, Kovai Kalaimagal College of Arts and Science, India, [mala.kkcas@gmail.com](mailto:mala.kkcas@gmail.com)

Multi-object tracking in video sequences presents significant challenges due to object variability, occlusions, and background clutter. Traditional approaches often fail to effectively model the intricate temporal dependencies and spatial interactions in video data. To overcome these challenges, this research introduces Multi Object Detection Tracking (MOD-Tracking), a novel deep learning framework that combines the strengths of Recurrent Neural Networks (RNN) and transformers. The model utilizes RNN to capture short-term temporal dependencies between frames and transformers to model long-term dependencies and global spatial context. Additionally, MOD-Tracking predicts bounding box regressions for precise localization of tracked objects, enhancing accuracy and robustness. A dynamic exponential decay learning rate schedule is incorporated to improve training efficiency and generalization, with the decay rate adaptively adjusted based on model performance. This ensures optimal learning throughout the training process and effective handling of complex tracking scenarios. Extensive experiments on challenging video datasets demonstrate the proposed model's superior capability to accurately track multiple objects, even amidst occlusions and cluttered backgrounds, setting a new benchmark for multi-object tracking tasks.

**Keywords:** Multi object tracking, Deep learning, Convolutional neural network, Recurrent neural network, Transformer, Dynamic decay learning rate, Bounding box regression, coordinate prediction.

## 1. Introduction

Multi-object tracking (MOT) in video sequences is a fundamental problem in computer vision with widespread applications ranging from surveillance systems and autonomous driving to sports analysis and robotics. The objective of MOT is to detect and consistently track multiple objects in a scene across consecutive frames, while maintaining the identities of these objects throughout their movements. This task is inherently challenging due to several factors, including object occlusions, varying lighting conditions, complex object dynamics, and the

presence of similarly appearing objects. Furthermore, as the number of objects increases or the scene becomes more cluttered, the tracking problem becomes significantly more difficult. Over the years, several approaches have been proposed to tackle MOT, with a majority relying on a combination of object detection and data association methods. Early approaches typically involved the application of Kalman filters, particle filters, or optical flow techniques to predict object trajectories.

However, recent advancements in deep learning and computer vision have revolutionized MOT by enabling more accurate detection and association of objects across frames. Modern methods leverage deep neural networks for robust object detection, appearance modeling, and feature extraction, alongside graph-based algorithms for data association, where objects are represented as nodes, and potential correspondences between them as edges. The increasing demand for real-time applications, such as autonomous navigation and video surveillance, also necessitates the development of computationally efficient MOT algorithms capable of processing high-resolution video streams with minimal latency. While several state-of-the-art methods offer impressive performance on standard benchmarks, they often face trade-offs between accuracy, speed, and robustness in real-world environments.

This research introduces a novel approach to multi-object tracking, leveraging the strengths of RNN and transformers to overcome the limitations of traditional methods. The proposed hybrid deep learning model focuses on improving the robustness of tracking by capturing both short-term temporal dependencies through RNN and long-term dependencies and global context with transformers. A dynamic exponential decay learning rate schedule is also incorporated to enhance training efficiency, adapting the learning rate based on model performance. This ensures the model is optimized to handle challenges such as occlusions, background clutter, and complex object interactions. Extensive experimentation demonstrates the model's superior ability to accurately track multiple objects in crowded and dynamic scenes, minimizing identity switches while maintaining computational efficiency. By addressing these key challenges, this research advances the field of multi-object tracking, providing more reliable and precise tracking performance for real-world applications.

## **2. LITERATURE REVIEW**

Wu, D., et al., 2023 introduced a novel task, Referring Multi-Object Tracking (RMOT), utilizing language expressions as semantic cues to guide multi-object tracking predictions. They presented the Refer-KITTI benchmark with 818 expressions and developed TransRMOT, a transformer-based model, which outperformed other approaches. This work marked a significant advancement in predicting varying numbers of referent objects in videos. Xiao, C., et. al., 2024 presented MotionTrack, an innovative motion-based tracker designed to improve object tracking through a learnable motion predictor that relies only on trajectory information. By leveraging self-attention and dynamic MLP layers, the method enhanced the modeling of temporal dynamics, delivering state-of-the-art performance on challenging datasets like Dancetrack and SportsMOT. This approach effectively tackled the difficulties of tracking objects with similar appearances and diverse motion patterns. Weng, X., et. al., 2020 proposed two techniques to improve discriminative feature learning in 3D multi-object tracking. They introduced a Graph Neural Network (GNN) to enhance feature interaction

across objects and a joint feature extractor to combine 2D and 3D modalities. Their method reduced object confusion and improved feature discrimination. Extensive evaluation demonstrated state-of-the-art performance on KITTI and nuScenes benchmarks.

Brasso, G. et. al., 2020 tackled the challenge of applying learning methods to graph-based multiple object tracking within the tracking-by-detection paradigm. They proposed a fully differentiable framework based on Message Passing Networks (MPNs) to operate directly on the graph domain. Their method allowed for global reasoning over detection sets and applied learning to the data association step, beyond just feature extraction. This approach enhanced the classical network flow formulation of MOT. Zhang, Y., et. al., 2021 tackled the issue of competing tasks in multi-object tracking by introducing FairMOT, which strikes a balance between object detection and re-identification (re-ID) in a unified network. They observed that prior approaches favored detection at the expense of re-ID, creating a bias. By leveraging the anchor-free CenterNet architecture with detailed design optimizations, FairMOT ensured strong performance in both tasks, achieving state-of-the-art results and significantly enhancing detection and tracking accuracy on public datasets. Yu, Y., et. al., 2020 proposed Deformable Siamese Attention Networks (SiamAttn) to enhance visual object tracking. They addressed the limitations of traditional Siamese trackers by introducing a novel attention mechanism that computes deformable self-attention and cross-attention. This approach allowed for adaptive target template updates and rich contextual interdependencies between the template and search image. Additionally, a region refinement module improved tracking accuracy. Experiments across six benchmarks demonstrated that SiamAttn outperformed the baseline, achieving state-of-the-art results.

Wang, Y., et. al., 2021 proposed a joint multi-object tracking framework utilizing graph neural networks to optimize object detection and data association concurrently. Their GNN-based approach effectively captured spatial and temporal relationships between objects, improving feature learning. Comprehensive experiments on MOT15, MOT16, MOT17, and MOT20 datasets showcased the method's state-of-the-art performance in both detection and tracking tasks. Chu, P., et. al., 2023 introduced TransMOT, a method for multi-object tracking that utilizes graph transformers to model spatial-temporal interactions between objects efficiently. TransMOT arranges tracked object trajectories and detection candidates as sparse weighted graphs and processes them through specialized transformer encoder and decoder layers. This approach allows for robust association estimation from loosely filtered detection predictions. Evaluated on multiple benchmarks including MOT15, MOT16, MOT17, and MOT20, TransMOT achieved state-of-the-art performance across all datasets. Pang, J., et. al., 2021 introduced Quasi-Dense Similarity Learning to enhance object tracking by densely sampling hundreds of region proposals for contrastive learning. Their method, Quasi-Dense Tracking (QDTrack), effectively combines this similarity learning with existing detection techniques, avoiding the need for displacement regression or motion priors. QDTrack achieved impressive results, outperforming existing methods on various benchmarks, including MOT, BDD100K, Waymo, and TAO. Notably, it reached 68.7 MOTA at 20.3 FPS on MOT17 without using external training data, significantly improving MOTA and reducing ID switches on BDD100K and Waymo datasets.

Li, S., et. al., 2023 addressed the limitations of traditional multiple object tracking benchmarks by introducing open-vocabulary MOT, which evaluates tracking beyond predefined

categories. They developed OVTrack, an open-vocabulary tracker capable of tracking arbitrary object classes. The design of OVTrack incorporates vision-language models for classification and association through knowledge distillation, along with a data hallucination strategy for robust feature learning using denoising diffusion probabilistic models. This approach resulted in a data-efficient tracker that achieved state-of-the-art performance on the large-scale TAO benchmark, trained solely on static images. Wang, X., et. al., 2021 introduced tracking by natural language specification, focusing on locating objects in videos using semantic descriptions instead of bounding boxes. They created the TNL2K benchmark, featuring 2,000 annotated video sequences to evaluate this new approach. The benchmark includes challenges like adversarial samples and modality switch, along with a strong baseline method based on an adaptive local-global search scheme, aimed at enhancing research in natural language-guided tracking. Cao, J., et. al., 2023 presented Observation-Centric SORT (OC-SORT), an enhancement of traditional Kalman filter methods for multi-object tracking. Recognizing that linear motion assumptions can lead to significant errors during prolonged occlusions, they proposed using object observations to compute a virtual trajectory that mitigates error accumulation. This approach allows for more effective error correction over time, improving robustness during occlusion and non-linear motion. OC-SORT maintains simplicity and real-time performance, achieving over 700 FPS on a single CPU while setting state-of-the-art results on various datasets, including MOT17, MOT20, and DanceTrack.

Chen, B., et. al., 2022 developed SimTrack, a Simplified Tracking architecture using a transformer backbone for efficient joint feature extraction and interaction. By serializing input images and employing a foveal window strategy to reduce information loss, they eliminated the need for complex interaction modules. SimTrack achieved 2.5% and 2.6% AUC gains on the LaSOT and TNL2K benchmarks, showing competitive performance against specialized tracking algorithms without intricate designs. Zeng, F., et. al., 2022 introduced MOTR, a method for multiple-object tracking that enhances temporal modeling by incorporating a "track query" mechanism, which updates tracked instances frame-by-frame. This approach improves upon traditional motion and appearance-based heuristics by allowing end-to-end temporal exploitation. MOTR achieved a 6.5% improvement over ByteTrack on the HOTA metric and outperformed concurrent methods like TrackFormer and TransTrack on the MOT17 dataset, positioning it as a strong baseline for future research in transformer-based tracking. Zhang, Y., et. al., 2022 introduced ByteTrackV2, a multi-object tracking method that employs a hierarchical data association strategy to effectively manage low-score detection boxes, reducing object missing and fragmented trajectories. They also implemented a motion prediction strategy using a Kalman filter to handle abrupt movements in 3D scenarios. ByteTrackV2 achieved top rankings on the nuScenes 3D MOT leaderboard, with 56.4% AMOTA for camera and 70.1% for LiDAR modalities, and its nonparametric design allows easy integration with various detectors for practical applications. Ma, F., et. al., 2022 proposed the Unified Transformer Tracker (UTT) to handle both Single Object Tracking (SOT) and multiple object tracking within a single framework. UTT uses a track transformer to exploit correlations between target and tracking frame features, allowing for effective localization. The model supports end-to-end training by alternately optimizing SOT and MOT objectives, leveraging large-scale tracking datasets. Experimental results across multiple benchmarks showed that UTT successfully addresses tracking challenges for both SOT and MOT tasks. Cai, J., et. al., 2022 developed MeMOT, an online tracking algorithm that integrates object

detection and data association using a large spatio-temporal memory to link objects over long periods. The model comprises three Transformer-based modules: Hypothesis Generation for object proposals, memory encoding for extracting relevant information, and memory decoding for simultaneous detection and association. MeMOT demonstrated competitive performance on standard MOT benchmark datasets.

Meinhardt, T., et. al., 2022 developed TrackFormer, a multi-object tracking method using an encoder-decoder transformer architecture. It formulates tracking as a frame-to-frame set prediction problem, employing attention mechanisms to evolve track predictions through video sequences. The model simplifies the process by using static and identity-preserving track queries, achieving state-of-the-art performance on MOT17 and MOTS20 without complex graph optimization.

Sun, P., et. al., 2020 proposed TransTrack, a transformer-based method for multiple object tracking that combines detection and tracking into a single step. It uses object features from previous frames as queries and learned object queries for new detections. TransTrack achieved MOTA scores of 74.5% on MOT17 and 64.5% on MOT20, demonstrating its efficiency and competitiveness in the field. Luo, W., et al., 2021 presented the first comprehensive review of multiple object tracking, emphasizing its significance in both academic and commercial contexts despite challenges such as sudden appearance changes and severe occlusions. They analyzed recent advancements across various facets of MOT, classifying methods, assessing fundamental principles, and summarizing experimental results on widely-used datasets for detailed comparisons. They also addressed critical issues within the field and suggested potential directions for future research, thereby addressing a significant gap in the existing literature.

Existing multi-object tracking methods struggle with integrating multimodal data, limiting performance in complex environments. They often rely on predefined categories, reducing adaptability to novel objects. Balancing detection and re-identification also remain challenging, leading to tracking biases and identity switches, especially during occlusions and rapid movements. These limitations underscore the need for more versatile, adaptive models. This research aims to address the limitations of existing multi-object tracking methods by developing and evaluating a novel hybrid deep learning architecture, the Multi-Object Detection and Tracking model (MOD-Tracking), to enhance accuracy and robustness in video-based tracking. MOD-Tracking leverages the strengths of recurrent neural networks for capturing short-term temporal dependencies alongside transformers for modeling long-term dependencies and global context. By effectively handling challenges like occlusions, background clutter, and tracking biases, this study seeks to demonstrate that MOD-Tracking can significantly outperform traditional methods, providing a more adaptive and resilient solution for real-world tracking scenarios. The key contributions are:

- The integration of RNN allows for effective handling of short-term dynamics between frames, ensuring smooth and accurate tracking of object movements.
- The use of Transformers enhances the model's capability to understand complex interactions over extended periods, facilitating better contextual awareness and reducing identity switches.
- A dynamic learning rate schedule is incorporated, optimizing training efficiency and ensuring

the model adapts effectively to the specific challenges of the tracking task.

This paper is organized into five sections. The introduction outlines the multi-object tracking problem, discussing its significance in various applications such as surveillance, autonomous driving, and robotics. It presents the MOD-Tracking model as an innovative hybrid deep learning architecture developed to enhance tracking accuracy and resilience by effectively addressing challenges such as occlusions and background clutter. The literature review analyzes recent advancements in multi-object tracking methods, highlighting the limitations of traditional approaches and the need for innovative models that can handle complex temporal dependencies and interactions. The methodology section details the architecture of MOD-Tracking, explaining the integration of RNN and transformers, along with the implementation of a dynamic learning rate schedule. In the results and analysis section, experimental findings are presented, showcasing the model's superior performance compared to existing methods. The analysis discusses the strengths and limitations of the proposed approach, along with relevant performance metrics. Finally, the conclusion summarizes the key insights gained, assesses the model's effectiveness, and identifies potential avenues for future research to enhance multi-object tracking capabilities further.

### **3. METHODOLOGY**

The proposed multi-object tracking framework processes input video frames by feeding them into a series of convolutional layers to extract spatial features from each frame. These features are then passed through a Long Short-Term Memory (LSTM) layer to capture the temporal dependencies between frames. The LSTM layer is responsible for tracking objects over time by learning the motion patterns and relationships between successive frames. The output of the LSTM is then processed by a multi-head self-attention mechanism which enhances the model's ability to focus on critical interactions between objects across different frames. This self-attention mechanism is followed by an Add & Norm layer to stabilize training and maintain effective gradient flow. The temporal and spatial features are then passed through additional convolutional layers and max-pooling layers to further refine the extracted information. The final output is flattened and passed through fully connected dense layers to generate the final predictions for bounding box coordinates and object classifications.

A dynamic learning rate schedule is employed to optimize the model, with the learning rate adjusted based on performance during training. The model uses L2 loss for bounding box regression and categorical cross-entropy for object classification. This combination of convolutional layers, LSTM, and self-attention mechanisms enables the model to robustly track multiple objects across frames, even under challenging conditions such as occlusions and complex backgrounds. Figure 1 illustrates the flow of the proposed MOD-Tracking model, demonstrating its systematic approach from object detection to identity tracking.



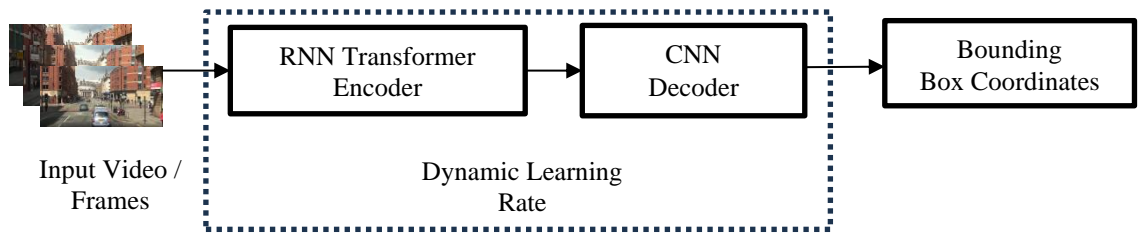


Figure 1: Flow of Proposed MOD-Tracking Model

### 3.1 RNN Transformer Encoder

In the encoder of the model, the input video frame undergoes a series of spatial and temporal processing steps to extract important features. The first layers are convolutional and pooling layers which focus on capturing spatial patterns and features from the input frame, such as edges, textures, and object shapes. These layers work as feature extractors, transforming the raw pixel data into a higher-level representation. After spatial features are extracted, the feature map is flattened and reshaped into a format that can be fed into the LSTM layer. The LSTM is responsible for learning the temporal dependencies across frames in the video sequence, understanding how objects move and interact over time. This allows the model to learn the motion and trajectory of objects, which is crucial for multi-object tracking. Additionally, a multi-head attention mechanism is used after the LSTM to focus on different parts of the sequence, allowing the model to consider relationships between frames and effectively track multiple objects simultaneously. Layer normalization is applied during this process to stabilize the learning and enhance the efficiency of training. By the end of the encoder, the video frames are encoded into a temporal representation that includes both the spatial features of objects and the motion context over time.

### 3.2 CNN Decoder

In the decoder, the processed information from the encoder is further refined to make the final predictions. The output from the encoder, which has encoded both the spatial and temporal information, is passed through additional layers to further refine the spatial features and adapt them to a form suitable for predicting the bounding box coordinates. These layers help the model learn to understand the relationship between the moving objects and their appearance over time. The output is then flattened again and passed through several dense layers, where the model applies non-linear transformations to generate the final output. The dense layers ultimately predict the bounding box coordinates, which represent the positions of objects in the video frames. These coordinates are crucial for tracking the objects as they move across the sequence. Through training, the model learns to map the encoded spatial and temporal features to the corresponding bounding box coordinates, allowing it to accurately track and predict the positions of multiple objects in future frames.

### 3.3 Dynamic Decay Learning Rate

A dynamic decay learning rate is applied during the training process to optimize performance and is combined with the Adam optimizer. Using an exponential decay strategy, the learning rate starts high for faster convergence during initial training and gradually decreases as training progresses. The Adam optimizer adjusts learning rates individually for each parameter based

on first and second moments of gradients, ensuring stable and efficient convergence. This approach helps fine-tune the model by avoiding overfitting and adapting to complex patterns in the data. Performance metrics such as tracking accuracy or loss are monitored to dynamically adjust the learning rate and ensure the model continues improving even when performance stagnates. This combination of a dynamic decay learning rate and the Adam optimizer enhances the model's robustness and consistency in handling complex multi-object tracking tasks. The learning rate at each epoch is calculated using the formula:

$$\eta_{\text{new}} = \eta_0 \cdot d^{\left(\frac{\text{epoch}}{s}\right)} \cdot p \quad (1)$$

Where,  $\eta_0$  be the initial learning rate,  $\eta_{\text{new}}$  be the new learning rate,  $d$  be the dynamic decay rate,  $p$  be the performance factor, epoch be the current training epoch,  $s$  be the decay steps. Here, the decay rate dynamic is given by:

$$d = \frac{1}{1 + \eta} \quad (2)$$

Where,  $\eta$  be the current learning rate. This introduces a gradual reduction in the learning rate, which slows down as training progresses to fine-tune the model. Additionally, the performance factor adjusts the learning rate based on the difference between the current validation performance and the best observed performance which is given by.

$$p = 1 - \text{val\_perf} - \text{best\_val\_perf} \quad (3)$$

Where,  $p$  be the performance factor,  $\text{val\_perf}$  be the validation performance,  $\text{best\_val\_perf}$  be the best validation performance so far. This factor helps in maintaining a higher learning rate when the model is performing well, while reducing it if the model's validation performance starts to deviate. The result is an adaptive learning schedule that improves convergence by balancing exploration in the early stages of training and careful fine-tuning as the model approaches optimal performance.

---

Algorithm – Dynamic\_decay\_learningrate ()

---

Update learning rate dynamically:

$$\text{decay\_rate\_dynamic} = \frac{1}{1 + \text{learning\_rate}}$$

$$\text{performance\_factor} = 1 - |\text{validation\_performance} - \text{best\_validation\_performance}|$$

$$\text{new\_learning\_rate} = \text{initial\_learning\_rate} \times \left( \text{decay\_rate\_dynamic} \right)^{\frac{\text{epoch}}{\text{decay\_steps}}} \times \text{performance\_factor}$$


---

### 3.4 Multiple Object Tracking





The proposed multiple object detection and tracking architecture begins with an input layer designed to handle sequences of video frames. The initial processing uses two convolutional layers each followed by max-pooling layers to extract spatial features from each frame. These layers progressively reduce the spatial dimensions while learning hierarchical patterns critical for object detection. The convolutional layers employ the ReLU activation function to capture



complex spatial features effectively. After the initial convolutional layers the output is flattened and passed through a Long Short-Term Memory (LSTM) layer which captures temporal dependencies across frames. The LSTM layer is crucial for tracking objects over time by learning the motion patterns and relationships between successive frames. The sequence of features from the LSTM is then passed through a multi-head self-attention mechanism which enables the model to focus on different parts of the sequence simultaneously and enhance the tracking ability. This attention mechanism is paired with an Add & Norm layer to apply residual connections and layer normalization stabilizing training and ensuring effective gradient flow.

Further convolutional layers are applied to refine the spatial features followed by max-pooling layers to progressively downsample the data. These convolutional layers help extract detailed and hierarchical spatial features that are important for accurate object detection. After the final convolutional layers the output is flattened again and passed through a series of fully connected dense layers with 256 128 and 64 units to further process the features and generate predictions the model predicts the bounding box coordinates [x\_min, y\_min, x\_max, y\_max] and class probabilities for each detected object. The model is trained using the Adam optimizer with a dynamic decay learning rate that adjusts based on performance during training. L2 loss is used for bounding box regression to ensure precise localization while categorical cross-entropy is used for object classification. The training process spans 50 epochs allowing the model to learn both temporal features from the LSTM and spatial features from the convolutional layers. These combined capabilities enable efficient and accurate multiple object detection and tracking. Table 1 showcase the output coordinates obtained for few sample frames and figure 2 shows the architecture design for the MOD-Tracking model

Table 1: Results of few samples frames

| Image   | Frame id | Object id | Object coordinates | Predicted coordinates |
|---|----------|-----------|--------------------|-----------------------|
|  | 1        | 1         | [287,300,220,100]  | [290,310,225,104]     |
|  | 1        | 2         | [128,300,300,120]  | [132,314,302,122]     |
|  | 2        | 1         | [330,273,220,100]  | [336,281,227,109]     |
|  | 2        | 2         | [146,275,300,120]  | [151,278,304,124]     |

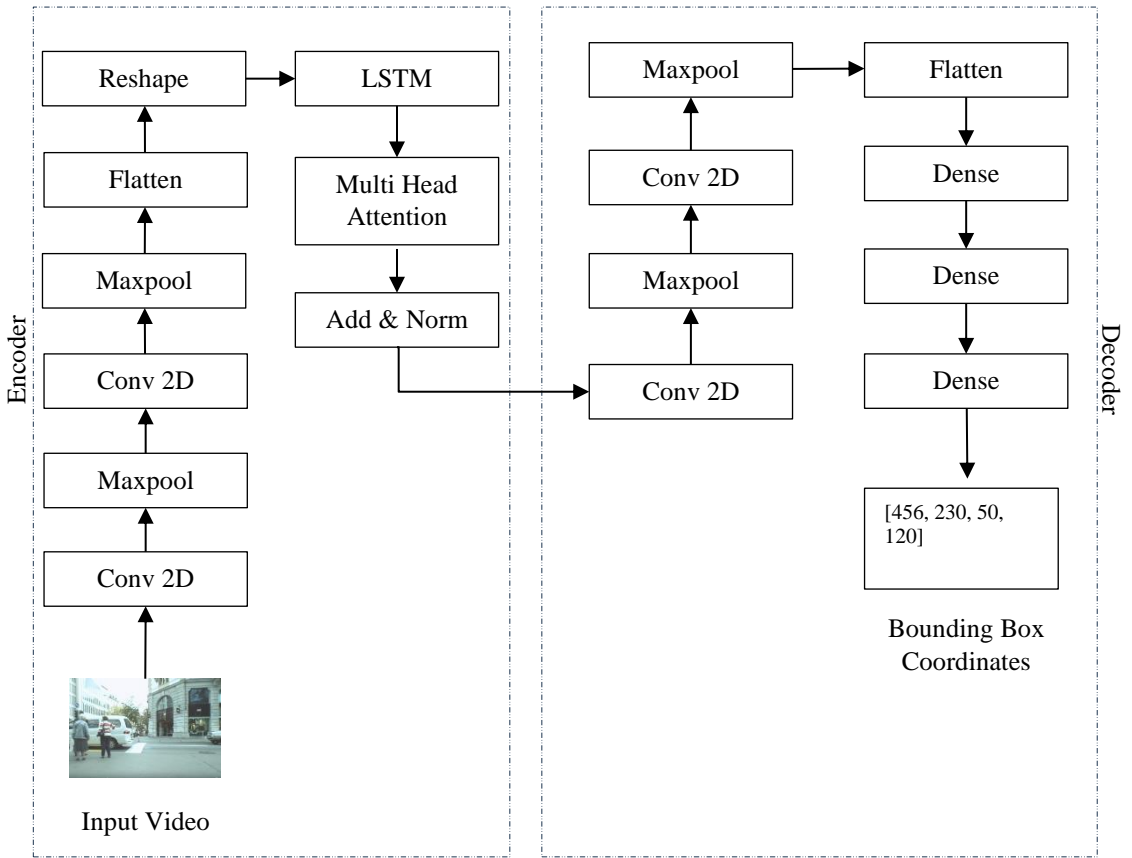


Figure 2: The architecture diagram of the multiple object tracking

#### Algorithm – MOD\_Tracking ()

##### 1. Preprocessing

1.1. Resizing:  
 Frame  
 Resize each input frame  $I_t$  to a fixed size (224, 224):

$$I_{t, \text{resized}} = \text{resize}(I_t, (244, 244))$$

##### 2. Model Architecture

###### 2.1 Input Layer:

Input sequence of frames  $X \in \mathbb{R}^{T \times 224 \times 224 \times 3}$ , where T is the number of frames.

###### 2.2. Convolutional Layers:

Apply the first two convolutional layers followed by max-pooling to extract spatial features from the frames.

$$H_{\text{conv1}} = \text{ReLU}(\text{Conv2D}(X, 64, (3, 3)))$$

$$H_{\text{pool1}} = \text{MaxPool}(H_{\text{conv1}}, (2,2))$$

$$H_{\text{conv2}} = \text{ReLU}(\text{Conv2D}(X, 128, (3,3)))$$

$$H_{\text{pool2}} = \text{MaxPool}(H_{\text{conv2}}, (2,2))$$

### 2.3. Flatten Layer:

Flatten the output from the last pooling layer.

$$H_{\text{flat1}} = \text{Flatten}(H_{\text{pool2}})$$

### 2.4. LSTM Layer:

Apply LSTM with 128 units to capture temporal dependencies across frames.

$$H_{\text{LSTM}} = \text{LSTM}(H_{\text{flat1}})$$

### 2.5. Multi-Head Self-Attention:

Apply multi-head self-attention with 8 heads and a key dimension of 64 to learn relationships between frames.

$$H_{\text{att}} = \text{MultiHeadAttention}(H_{\text{LSTM}}, H_{\text{LSTM}})$$

### 2.6. Add & Norm Layer:

Add a residual connection and apply layer normalization.

$$H_{\text{add1}} = \text{LayerNorm}(H_{\text{LSTM}} + H_{\text{att}})$$

### 2.7. Convolutional Layers:

Apply further convolutional layers and max-pooling to refine the extracted spatial features.

$$H_{\text{conv3}} = \text{ReLU}(\text{Conv2D}(X, 256, (3,3)))$$

$$H_{\text{pool3}} = \text{MaxPool}(H_{\text{conv3}}, (2,2))$$

$$H_{\text{conv4}} = \text{ReLU}(\text{Conv2D}(X, 512, (3,3)))$$

$$H_{\text{pool4}} = \text{MaxPool}(H_{\text{conv4}}, (2,2))$$

### 2.8. Flatten Layer:

Flatten the output from the final convolutional layer to prepare for dense layer input.

$$H_{\text{flat2}} = \text{Flatten}(H_{\text{pool4}})$$

### 2.9. Fully Connected Layers:

Apply a series of dense layers to make the final prediction.

First Dense Layer:

$$H_{\text{dense1}} = \text{ReLU}(W_{\text{dense1}}H_{\text{flat2}} + b_{\text{dense1}})$$

Second Dense Layer:

$$H_{\text{dense2}} = \text{ReLU}(W_{\text{dense2}}H_{\text{dense1}} + b_{\text{dense2}})$$

Output Layer:

$$H_{\text{dense3}} = \text{ReLU}(W_{\text{dense3}}H_{\text{dense2}} + b_{\text{dense3}})$$

The output consists of bounding box coordinates  $[x_{\min}, y_{\min}, x_{\max}, y_{\max}]$  and class probabilities for each detected object.

### 3. Training

#### 3.1. Loss Function:

**Bounding Box Regression Loss:** Measures the error in predicted bounding box coordinates  $y_{\text{bbox}}$  compared to ground truth bounding box coordinates  $y^{\text{bbox}}$ . It uses the squared L2 norm:

$$L_{\text{bbox}} = \| y_{\text{bbox}} - y^{\text{bbox}} \|_2^2$$

**Classification Loss:** A Cross-Entropy loss comparing the predicted class probabilities  $y_{\text{class}}$  with the ground truth class labels  $\hat{y}_{\text{class}}$ :

$$L_{\text{class}} = \text{CrossEntropy}(y_{\text{class}}, \hat{y}_{\text{class}})$$

#### 3.2. Learning Rate Update:

Dynamic\_decay\_learningrate ()

#### 3.3. Optimization:

Update model parameters using the Adam optimizer:

$$\theta_{\text{new}} = \theta - \text{new\_learningrate} \nabla_{\theta} L$$

where  $\nabla_{\theta} L$  is the gradient of the loss with respect to the model parameters  $\theta$ .

#### 3.4. Iteration:

Repeat training for a specified number of epochs (e.g., 50 epochs).

### 4. Evaluation

#### 4.1. Prediction on Test Data:

Predict bounding boxes and class labels for test frames:

$$\hat{y}_{\text{test}} = \text{Predict}(X_{\text{test}}, \theta)$$

#### 4.2. Performance Assessment:

Assess model performance using metrics such as MOTA, IDF1, Precision, Recall.

---

## 4. EXPERIMENT RESULT AND ANALYSIS

### 4.1 Dataset Description

The MOT17 dataset is a comprehensive benchmark designed for evaluating multi-object  
*Nanotechnology Perceptions* Vol. 20 No. S15 (2024)

tracking algorithms, specifically targeting pedestrian tracking in real-world environments. It contains 14 video sequences filmed in diverse urban settings, capturing challenging scenarios such as crowded areas, frequent occlusions, camera motion, and varying lighting conditions. Each sequence is provided at two resolutions, and multiple detection results (from DPM, SDP, and FRCNN detectors) are also available for each. The dataset includes detailed ground truth annotations, where each frame is labelled with bounding boxes that indicate the location of pedestrians, along with unique object IDs that maintain consistent identities across frames. The ground truth data also includes visibility scores to account for occlusions and the proportion of the object visible in each frame. These annotations are stored in a structured format, listing frame number, object ID, bounding box coordinates (x, y, width, height), class label (pedestrian), and object visibility. With its rich diversity of scenes and precise ground truth labels, MOT17 serves as an essential resource for evaluating the robustness and accuracy of MOT systems in complex, real-world scenarios.

The VOT2016 dataset is a benchmark designed to evaluate object tracking algorithms accuracy and robustness across diverse challenges. It includes 60 annotated video sequences with various objects undergoing occlusions, scale changes, rotations, and illumination variations, providing a rigorous testbed for tracking models. Each frame in the sequences has precise rotated bounding box ground truth annotations, which capture the object's position and orientation, allowing detailed evaluation of localization performance. The dataset includes an evaluation protocol measuring both tracking accuracy (intersection over union with the ground truth) and robustness (failure counts, where the tracker loses the object). VOT2016 is widely used to benchmark and advance tracking algorithms in real-world scenarios, promoting the development of models capable of handling complex tracking conditions.

## 4.2 Experiment Setup

The experiments were conducted on a high-performance workstation equipped with an Intel Core i7-12700K processor clocked at 3.60 GHz with 32 GB of RAM and a 64-bit operating system using x64-based architecture running Windows 11. The implementation was performed using the Python programming language within the Anaconda integrated development environment. The TensorFlow and Keras libraries were utilized for model development and training while additional libraries such as NumPy, Pandas and Matplotlib were employed for data preprocessing and visualization. GPU acceleration was enabled using an NVIDIA GeForce RTX 3080 with 10 GB of VRAM to optimize training time and meet the computational demands of the MOD-Tracking framework. Both the proposed MOD-Tracking model and the comparison works were implemented and evaluated in this environment to ensure consistent and fair performance analysis.

## 4.3 Result Analysis

The proposed MOD-Tracking model combines RNN, a Transformer encoder, and a CNN-based decoder to capture both short- and long-term temporal dependencies for spatial object trajectory generation. Using a dynamic decay learning rate with exponential decay, it adapts based on tracking performance, ensuring resilient tracking under challenging conditions like occlusions and complex backgrounds. Performance of proposed MOD-Tracking model is compared with three existing work includes, work of Bai, H., et. al., 2021, Xu, Y., et. al., 2019 and Bochinski, E., et. al., 2018. Bai, H., et. al., 2021 made significant progress in Generic

Multiple Object Tracking (GMOT) by introducing GMOT-40, the first dense dataset for GMOT, consisting of 40 sequences across 10 object categories. They developed baseline algorithms and evaluated them alongside modified versions of popular MOT methods. The GMOT-40 benchmark is a valuable resource for future research in the field. Xu, Y., et. al., 2019 reviewed leading deep learning-based multi-object tracking methods, classifying them into three categories: feature enhancement, network embedding and end-to-end approaches. The authors demonstrated the effectiveness and robustness of these methods through benchmark comparisons, while also highlighting their limitations and proposing future research directions. Bochinski, E., et. al., 2018 tackled the problem of ID switches and fragmentations in multi-object tracking by incorporating visual single-object tracking when detections were absent. This strategy notably enhanced tracking performance while maintaining high speeds, outperforming state-of-the-art methods on the UA-DETRAC and VisDrone datasets. In multi-object tracking key metrics like MOTP, MOTA, IDF1, MT, and ML are used to assess tracking performance. These metrics evaluate the accuracy and reliability of tracking multiple objects over time.

- Multiple  
Object Tracking Precision (MOTP) measures the precision of predicted bounding boxes by calculating the average overlap between the predicted and ground truth boxes across all frames and objects. It is given by:

$$\text{MOTP} = \frac{\sum_{i,t} \text{IoU}_{i,t}}{\sum_t c_t}$$

where  $\text{IoU}_{i,t}$  is the Intersection over Union for object  $i$  at time  $t$ , and  $c_t$  is the number of correctly tracked objects at time  $t$ .

- Multiple  
Object Tracking Accuracy (MOTA) reflects the overall tracking accuracy by considering missed detections (false negatives), false positives, and identity switches. It is defined as:

$$\text{MOTA} = 1 - \frac{\sum_t (\text{FN}_t + \text{FP}_t + \text{ID}_{st})}{\sum_t \text{GT}_t}$$

Where  $\text{FN}_t$  is the number of false negatives,  $\text{FP}_t$  is the number of false positives,  $\text{ID}_{st}$  is the number of identity switches, and  $\text{GT}_t$  is the number of ground truth objects at time  $t$ .

- ID      F1  
Score (IDF1) evaluates the accuracy of identity preservation across frames, computed as the harmonic mean of ID precision and ID recall:

$$\text{IDF1} = 2 \times \text{IDTP} + \text{IDFP} + \text{IDFN} \times \text{IDTP}$$

where IDTP, IDFP and IDFN are the true positives, false positives, and false negatives in terms of identity matching.

- Mostly  
Tracked (MT) denotes the percentage of ground truth objects tracked for at least 80% of their lifespan:



$$MT = \frac{\text{Number of mostly tracked objects}}{\text{Total number of ground truth objects}} \times 100$$

Higher MT indicates more successful tracking.

• Mostly  
Lost (ML) represents the percentage of objects tracked for less than 20% of their lifespan:

$$ML = \frac{\text{Number of mostly lost objects}}{\text{Total number of ground truth objects}} \times 100$$

Lower ML suggests fewer objects are being lost during tracking.

The metrics discussed above MOTP, MOTA, IDF1, MT, and ML are used to compare MOD-Tracking with existing methods.

Table 2: Comparison of MOD-Tracking performance with and without proposed decay learning rate

| Learning rate                | MOTP   | MOTA   | IDF1   | MT     | ML     |
|------------------------------|--------|--------|--------|--------|--------|
| Static default learning rate | 73.93% | 78.24% | 79.16% | 15.82% | 26.59% |
| Proposed decay learning rate | 76.2%  | 80.7%  | 80.1%  | 17.42% | 24.8%  |

Table 2 shows that applying the proposed decay learning rate improves MOD-Tracking performance, especially under occlusions and complex backgrounds. Without decay learning rate the tracking accuracy gradually declines, underscoring the decay rate's impact on model robustness.

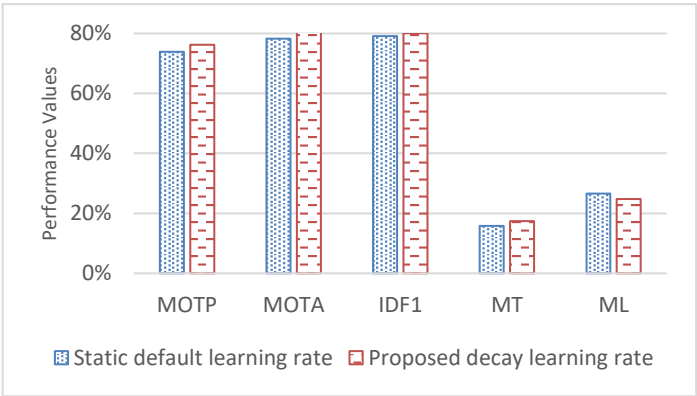


Figure 3: Graphical representation of proposed MOD-Tracking performance with and without proposed decay learning rate

Figure 3 shows that the proposed decay learning rate enhances MOD-Tracking accuracy, especially under challenging conditions. Without decay learning rate, accuracy declines over time, emphasizing the decay rate's stabilizing effect.

Table 3: Comparison of Proposed MOD-Tracking Performance Metrics with Existing Methods using MOT17 dataset

| Methods                      | MOTP   | MOTA   | IDF1   | MT     | ML     |
|------------------------------|--------|--------|--------|--------|--------|
| Proposed MOD-Tracking        | 76.2%  | 80.7%  | 80.1%  | 17.42% | 24.8%  |
| Bai, H., et. al., 2021.      | 75.16% | 80.60% | 79.30% | 16.12% | 25.18% |
| Xu, Y., et. al., 2019        | 75.8%  | 47.17% | 46.3%  | 15.96% | 26.77% |
| Bochinski, E., et. al., 2018 | 75.8 % | 42.6%  | 58.0 % | 14.82% | 25.9%  |

The table 3 presents a comparison of multiple object tracking performance across proposed MOD-Tracking and existing methods using MOT17 dataset based on three key metrics: MOTP, MOTA, IDF1, MT and ML. The proposed MOD-Tracking method achieved the highest scores across all metrics, with 76.2% MOTP, 80.7% MOTA, 80.1% IDF1, MT 17.42% and ML 24.8% indicating superior precision, overall accuracy, and strong identity preservation. This highlights the proposed MOD-Tracking method's effectiveness in maintaining both tracking accuracy and identity consistency compared to existing approaches.

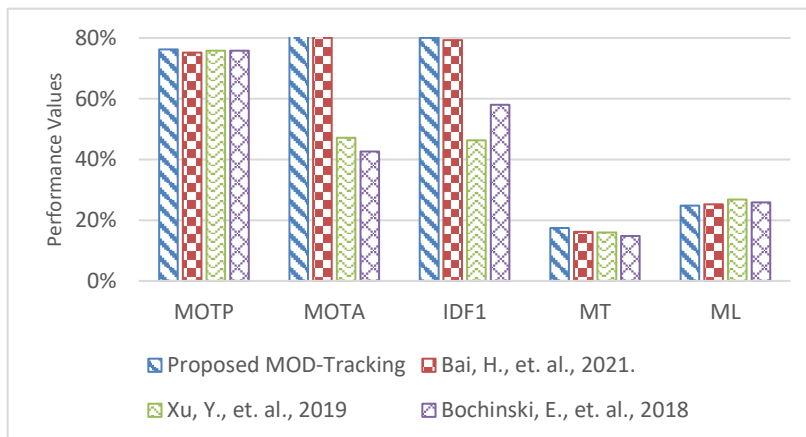


Figure 4: Graphical Representation of Proposed MOD-Tracking Performance Metrics with Existing Methods using MOT17 dataset

The figure 4 show the graphical representation of the proposed MOD-Tracking performance metrics alongside existing methods using MOT17 dataset provides a clear visual comparison of tracking effectiveness across different approaches. The graph showcases key metrics such as MOTP, MOTA, IDF1, MT and ML illustrating how the proposed method outperforms existing ones in all categories.

Table 4: Comparison of Proposed MOD-Tracking Performance Metrics with Existing Methods using VOT16 dataset

| Methods                      | MOTP   | MOTA   | IDF1   | MT     | ML     |
|------------------------------|--------|--------|--------|--------|--------|
| Proposed MOD-Tracking        | 75.30% | 80.10% | 79.29% | 16.02% | 23.86% |
| Bai, H., et. al., 2021.      | 74.12% | 79.80% | 78.40% | 15.90% | 24.06% |
| Xu, Y., et. al., 2019        | 74.70% | 45.15% | 45.80% | 14.12% | 25.50% |
| Bochinski, E., et. al., 2018 | 74.10% | 41.00% | 57.30% | 13.03% | 24.16% |

Table 4 presents a comparison of the proposed MOD-Tracking method against existing multi-object tracking approaches using the VOT16 dataset, showcasing its superior performance across key metrics such as MOTA, MOTP, and ID F1 score. The method's enhanced resilience to challenges like occlusions and rapid object movements highlights its effectiveness in maintaining tracking continuity and reducing identity switches. These findings establish MOD-Tracking as a leading solution in the field of multi-object tracking, raising the standards for real-world applications.

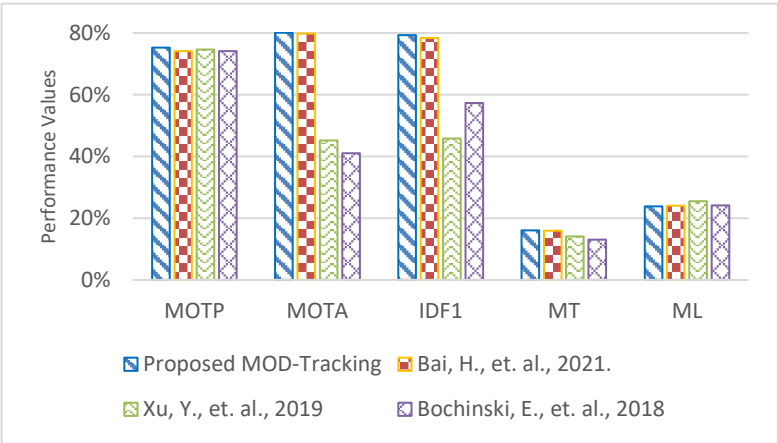


Figure 5: Graphical Representation of Proposed MOD-Tracking Performance Metrics with Existing Methods using VOT16 dataset

Figure 5 illustrates the performance metrics of the proposed MOD-Tracking method in comparison to existing multi-object tracking approaches using the VOT16 dataset. The graphical representation highlights MOD-Tracking's superior results across key metrics, such as MOTA, MOTP, and ID F1 score, effectively demonstrating its enhanced tracking capabilities. This visual comparison emphasizes the method's robustness and effectiveness in challenging tracking scenarios.

Table 5: Comparison of MOD-Tracking Performance Metrics for Different Datasets

| Dataset | MOTP   | MOTA  | IDF1  | MT     | ML    |
|---------|--------|-------|-------|--------|-------|
| MOT17   | 76.2%  | 80.7% | 80.1% | 17.42% | 24.8% |
| VOT16   | 75.99% | 80.1% | 79.8% | 16.92% | 25.1% |

Table 5 shows the comparison between MOD-tracking performance metrics across different datasets reveals variations in tracking precision, accuracy, and identity preservation. Metrics such as MOTA, MOTP, IDF1, MT, and ML provide insights into how well each tracker performs under different conditions, including object density, occlusions, and motion complexity.

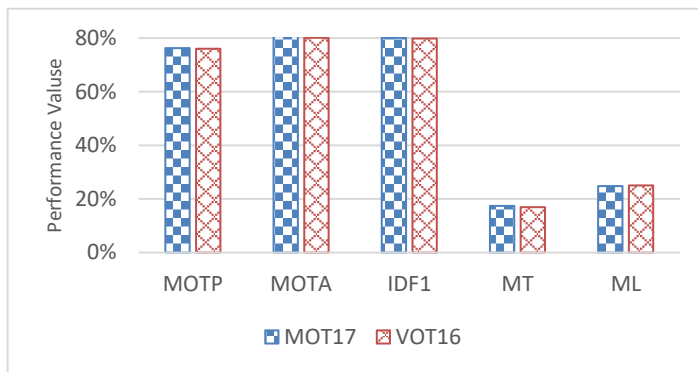


Figure 6: Graphical representation of MOD-Tracking Performance Metrics for Different Datasets

Figure 6 provides a visual overview of the MOD-Tracking model's performance metrics across various datasets, illustrating its effectiveness in handling diverse tracking challenges. The graphical representation highlights the model's accuracy, robustness, and adaptability, particularly in scenarios involving occlusions and background clutter.

## Discussion

The proposed MOD-Tracking method effectively addresses the limitations of existing multi-object tracking approaches, demonstrating significant advancements in tracking performance, particularly with the MOT17 dataset where it achieved notable scores, including 76.2% for MOTP, 80.7% for MOTA, and 80.1% for IDF1. The MOD-Tracking model outperformed existing methods due to its innovative hybrid architecture that effectively combines RNN, Transformer encoders, and CNN-based decoders. This unique integration allows the model to simultaneously capture short-term temporal dependencies and long-term spatial relationships, addressing limitations in traditional approaches. By leveraging RNNs, the model learns motion patterns and temporal continuity, while the Transformer encoders enhance its ability to focus on critical regions and objects in a sequence, even in the presence of occlusions or similar object appearances. The CNN-based decoder further refines spatial feature extraction, enabling accurate localization and classification of objects. A key contribution of the MOD-

Tracking method is its novel decay learning rate designed to adapt the training process dynamically. The exact purpose of this learning rate mechanism is to ensure stable and effective training, particularly in challenging scenarios such as rapid object movements, frequent occlusions, and highly dynamic backgrounds. By gradually reducing the learning rate as training progresses, the model avoids overfitting to noisy or less informative patterns while fine-tuning its parameters for precision in later stages. This approach significantly enhances the model's ability to generalize across complex tracking scenarios, where abrupt learning rate adjustments might otherwise destabilize the training process. The significant application of this learning rate strategy lies in maintaining the model's robustness in real-world environments where tracking conditions are highly variable. For instance, in dense object interactions or scenes with objects exhibiting similar appearances, the dynamic decay learning rate ensures that the model continues to learn nuanced motion and spatial features without overreacting to transient tracking errors. Additionally, the mechanism prevents premature convergence, allowing the model to refine its understanding of intricate object trajectories over time. This contributes to superior metrics across critical performance indicators, not only improving tracking precision but also significantly reducing identity switches and fragmentations—common issues in current techniques.

These enhancements are especially crucial in real-world applications, where occlusions, rapid object movements, and complex environments often lead to tracking errors. By addressing these challenges, the MOD-Tracking method provides a more robust and reliable solution for accurate multi-object tracking, making it well-suited for complex scenarios such as traffic monitoring, surveillance systems, and sports analytics.

## 5. CONCLUSION

In conclusion, the MOD-Tracking model demonstrates a significant improvement in multiple object tracking performance by effectively addressing key challenges such as identity preservation and tracking accuracy. With its superior metrics in MOTP, MOTA, and IDF1, the proposed model not only outperforms existing methods but also showcases its robustness in maintaining consistent object identities even in complex scenarios. The findings highlight the model's potential for practical applications in various fields, such as surveillance and autonomous systems, where reliable object tracking is critical. Furthermore, the encouraging results suggest opportunities for future enhancements and adaptations of the MOD-Tracking model to tackle even more intricate tracking environments. Overall, this research contributes valuable insights to the evolving landscape of multi-object tracking and sets the stage for ongoing advancements in the field.

## References

1. Bai, H., Cheng, W., Chu, P., Liu, J., Zhang, K. and Ling, H., 2021. Gmot-40: A benchmark for generic multiple object tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 6719-6728).
2. Bochinski, E., Senst, T. and Sikora, T., 2018, November. Extending IOU based multi-object tracking by visual information. In 2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS) (pp. 1-6). IEEE.
3. Brasó, G. and Leal-Taixé, L., 2020. Learning a neural solver for multiple object tracking. In Proceedings *Nanotechnology Perceptions* Vol. 20 No. S15 (2024)

- of the IEEE/CVF conference on computer vision and pattern recognition (pp. 6247-6257).
4. Cai, J., Xu, M., Li, W., Xiong, Y., Xia, W., Tu, Z. and Soatto, S., 2022. Memot: Multi-object tracking with memory. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 8090-8100).
5. Cao, J., Pang, J., Weng, X., Khirodkar, R. and Kitani, K., 2023. Observation-centric sort: Rethinking sort for robust multi-object tracking. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 9686-9696).
6. Chen, B., Li, P., Bai, L., Qiao, L., Shen, Q., Li, B., Gan, W., Wu, W. and Ouyang, W., 2022, October. Backbone is all your need: A simplified architecture for visual object tracking. In European Conference on Computer Vision (pp. 375-392). Cham: Springer Nature Switzerland.
7. Chu, P., Wang, J., You, Q., Ling, H. and Liu, Z., 2023. Transmot: Spatial-temporal graph transformer for multiple object tracking. In Proceedings of the IEEE/CVF Winter Conference on applications of computer vision (pp. 4870-4880).
8. Li, S., Fischer, T., Ke, L., Ding, H., Danelljan, M. and Yu, F., 2023. Ovtrack: Open-vocabulary multiple object tracking. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 5567-5577).
9. Luo, W., Xing, J., Milan, A., Zhang, X., Liu, W. and Kim, T.K., 2021. Multiple object tracking: A literature review. Artificial intelligence, 293, p.103448.
10. Ma, F., Shou, M.Z., Zhu, L., Fan, H., Xu, Y., Yang, Y. and Yan, Z., 2022. Unified transformer tracker for object tracking. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 8781-8790).
11. Meinhardt, T., Kirillov, A., Leal-Taixe, L. and Feichtenhofer, C., 2022. Trackformer: Multi-object tracking with transformers. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 8844-8854).
12. Pang, J., Qiu, L., Li, X., Chen, H., Li, Q., Darrell, T. and Yu, F., 2021. Quasi-dense similarity learning for multiple object tracking. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 164-173).
13. Sun, P., Cao, J., Jiang, Y., Zhang, R., Xie, E., Yuan, Z., Wang, C. and Luo, P., 2020. Transtrack: Multiple object tracking with transformer. arXiv preprint arXiv:2012.15460.
14. Wang, X., Shu, X., Zhang, Z., Jiang, B., Wang, Y., Tian, Y. and Wu, F., 2021. Towards more flexible and accurate object tracking with natural language: Algorithms and benchmark. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 13763-13773).
15. Wang, Y., Kitani, K. and Weng, X., 2021, May. Joint object detection and multi-object tracking with graph neural networks. In 2021 IEEE international conference on robotics and automation (ICRA) (pp. 13708-13715). IEEE.
16. Weng, X., Wang, Y., Man, Y. and Kitani, K.M., 2020. Gnn3dmot: Graph neural network for 3d multi-object tracking with 2d-3d multi-feature learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 6499-6508).
17. Wu, D., Han, W., Wang, T., Dong, X., Zhang, X. and Shen, J., 2023. Referring multi-object tracking. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 14633-14642).
18. Xiao, C., Cao, Q., Zhong, Y., Lan, L., Zhang, X., Luo, Z. and Tao, D., 2024. Motiontrack: Learning motion predictor for multiple object tracking. Neural Networks, 179, p.106539.
19. Xu, Y., Zhou, X., Chen, S. and Li, F., 2019. Deep learning for multiple object tracking: a survey. IET Computer Vision, 13(4), pp.355-368.
20. Yu, Y., Xiong, Y., Huang, W. and Scott, M.R., 2020. Deformable siamese attention networks for visual object tracking. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 6728-6737).
21. Zeng, F., Dong, B., Zhang, Y., Wang, T., Zhang, X. and Wei, Y., 2022, October. Motr: End-to-end multiple-object tracking with transformer. In European Conference on Computer Vision (pp. 659-675). Cham: Springer Nature Switzerland.
22. Zhang, Y., Sun, P., Jiang, Y., Yu, D., Weng, F., Yuan, Z., Luo, P., Liu, W. and Wang, X., 2022, October. Bytetrack: Multi-object tracking by associating every detection box. In European conference on computer vision (pp. 1-21). Cham: Springer Nature Switzerland.
23. Zhang, Y., Wang, C., Wang, X., Zeng, W. and Liu, W., 2021. Fairmot: On the fairness of detection and re-identification in multiple object tracking. International journal of computer vision, 129, pp.3069-3087.