Journal of Theoretical and Applied Information Technology
15th February 2026. Vol.104. No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

ENHANCING AUTONOMOUS DRONE DELIVERY SYSTEMS: A HYBRID CNN-LSTM APPROACH FOR ROBUST OBJECT TRACKING IN DYNAMIC ENVIRONMENTS

ANNAPURNA GUMMADI[1], RAVINDRA CHANGALA[2], PULLAIAH PINNIKA[3], B SRIVANI[4], VIJAYAKUMARI RODDA[5], Dr. N NEELIMA[6], R KIRTHIGA[7*]

[1]Department of CSE (Data Science), CVR College of Engineering, Hyderabad, Telangana, India
[2]Department of CSE, Guru Nanak Institutions Technical Campus, Hyderabad, Telangana, India
[3]Department of CSE(AI&ML), Kallam Haranadhareddy Institute of Technology, Guntur, Andhra Pradesh, India
[4]Department of IT, VNR Vignana Jyothi Institute of Engineering & Technology, Hyderabad, Telangana, India
[5]Department of Computer Science, Krishna University, Machilipatnam, Andhra Pradesh, India
[6]Department of CSE, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Andhra Pradesh, India
[*7]Department of Artificial Intelligent and Machine Learning, K Ramakrishnan College of Engineering, Samayapuram, Tamil Nadu, India
E-mail: [1]gummadiannapurna@gmail.com, [2]changalaravindra@gmail.com, [3]pullaiah.531@gmail.com, [4]srivani_b@vnrvjiet.in, [5]vijayakumari28@gmail.com, [6]gandhamneelu@gmail.com, [*7]smjrkrose@gmail.com

ABSTRACT

Strong object tracking is needed in autonomous drone delivery to manage changing surroundings appropriately. This research introduces an innovative method that uses CNNs to extract features and LSTM networks to predict the sequence in which objects appear in videos. The primary goal is to enhance the tracking system's reliability while objects move and are hidden and when the environment is unpredictable. We measured the precision, success rate, real-time FPS, and occlusion handling of the method with the help of the MOT Challenge and our custom drone flight dataset. Experiments found that the Proposed Model is more accurate than SiamFC, DeepSORT, and CNN + Kalman Filter, with a precision of 94% and a success rate of 92%, and can operate at 30 FPS in live tests. The model is trusted for uncrewed drone operations because it can overcome occlusions and accurately restore missing parts. This approach improves how autonomous drones navigate, making them useful for logistics, keeping watch, and handling emergencies. The primary contribution of this work lies in the development of a hybrid CNN–LSTM architecture integrated with sensor fusion for robust real-time object tracking. The proposed approach advances existing solutions by improving tracking accuracy, occlusion recovery, and real-time performance in dynamic drone delivery environments.

Keywords: *Autonomous Drones, Object Tracking, Deep Learning, Convolutional Neural Networks (CNNs), Long Short-Term Memory (LSTM), Sensor Fusion*

1. INTRODUCTION

In recent years, UAV technology, machine learning, and computer vision have been behind the increased attention on autonomous drones for last-mile shipping. With these systems, there is potential to lower the time needed for deliveries, cut down on costs, and support sustainability. A significant challenge in autonomous drone navigation is knowing where to find objects in the air. The safety, precision, and reliability of drone delivery are mainly guaranteed by tracking objects moving in busy surroundings [1], [2].

A drone needs to do object tracking to perform duties such as delivering packages, tracking, and keeping an eye on roads. The main difficulty is designing systems that follow targets precisely as they encounter obstacles, sudden motions, and situations where something covers the target from view. Before, popular tracking methods such as Kalman Filters, Mean-Shift, and Particle Filters worked well but struggled when objects or their motions were hidden or fast [3], [4]. Since these approaches depend on fixed forecasts for object motion, they rarely work in real-world applications where something is moving very fast [5].

Thanks to improvements in deep learning, object tracking now covers complicated settings that were difficult to handle. CNNs and RNNs, in particular, have made significant progress in helping to detect and follow objects over time. CNNs help to identify many valuable qualities in images. They are especially effective for spatial processing, while LSTM networks within RNNs take the lead in predicting where objects will be found next in a sequence, concerning previous times [6], [7]. When the two approaches are connected, the system shows clear improvements in how well targets are followed, especially when objects move unpredictably [8], [9].

For autonomous drone systems to be reliable, real-time tracking must consider video data and input from multiple sensors. Because sensors like cameras, LiDAR, and GPS provide different data, merging this data with sensor fusion is popular because it improves object location and tracking when visibility is reduced or lighting is poor [10], [11]. Magic Bench is improved when sensors are added because they help the system manage occlusions and items that are being moved around, both common issues during deliveries [12], [13].

To resolve these problems, this research introduces a new approach that uses CNNs to identify features and LSTMs to track objects throughout the sequence. Developing architecture that can provide reliable, timely tracking when the environment changes is the work's main aim, and this is crucial for the self-navigation of drones. Sensor fusion is used in this research to enhance the accuracy and robustness of tracking objects in video streams. The upcoming sections will discuss the methods, experimental findings, and ways this work could be used in real drone delivery.

The selection of object tracking as the core research focus was motivated by its critical role in ensuring safety, reliability, and autonomy in drone delivery systems operating in dynamic environments. Existing tracking approaches struggle under conditions involving rapid motion, occlusions, and environmental variability, which are common in real-world drone deployments. Addressing these challenges requires a robust learning framework capable of capturing both spatial appearance and temporal motion characteristics, thereby justifying the focus of this study.

The study addresses the following research questions:

(i) How can spatial and temporal learning be combined to improve object tracking in drones?
(ii) What is the impact of sensor fusion on tracking robustness under occlusion?
(iii) Can real-time performance be maintained without sacrificing accuracy?

The next part of this paper examines studies in autonomous drone object tracking that use deep learning and sensor fusion. In Section 3, the approach is explained, together with the CNN-LSTM model, the datasets chosen, and the sensor fusion methods. Section 4 shows the findings and comparisons with other models. At the end, Section 5 gathers all the results, discusses the limitations, and shares what research could be done next.

2. RELATED WORK

Due to its potential to make autonomous drones more accurate and reliable in changing situations, object tracking with deep learning has attracted attention. We covered traditional and recent deep learning approaches earlier. Still, a larger body of work explores the problems in object tracking for UAVs, paying attention to speed, sensor integration, and adaptability. We examine and discuss current research in deep learning-based object tracking for drone systems, bringing essential progress and approaches to light.

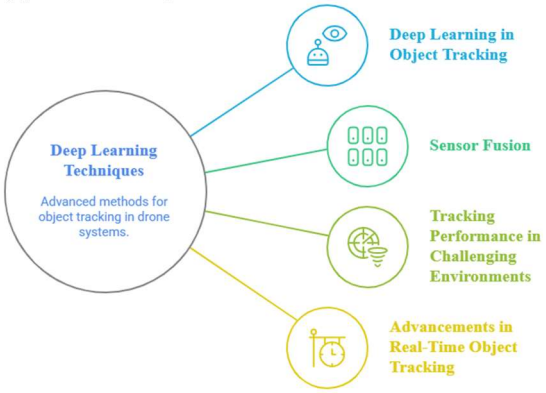

Figure 1: Exploring Deep Learning Techniques for Drone Tracking

Figure 1 shows the most important deep learning techniques in drone object tracking, focusing on what helps these systems improve. The leading theory of "Deep Learning Techniques" is to show how advanced methods can help drones track objects more accurately. Surrounding the main element, there are four other topics: "Deep Learning in Object Tracking" focuses on applying deep learning for tracking, "Sensor Fusion" combines sensor data to raise tracking accuracy, "Tracking Performance in Challenging Environments" explores deep learning solutions to overcome common roadblocks and "Advancements in Real-Time Object Tracking" touches on better results for real-time tracking. All

three work together to improve the efficiency of autonomous drone delivery.

2.1 Deep Learning in Object Tracking for Autonomous Drones

Much research involves using CNNs and RNNs to increase the accuracy of drone-based object tracking. Zhang et al. described in [14] how Siamese networks for monitoring have led to significant changes in object detection and tracking. With the Siamese network architecture, the system mainly learns how to match frames to track a target when the system has only come across it once. These models handle well if objects are hidden in the image or their appearance is very different than usual. Still, the system can perform poorly in real-time when the background is moving and objects are changing speed and direction [15], [16].

To overcome timing challenges and improve the system's reliability, Ullah et al. [17] designed an object-tracking system that pairs deep learning with features related to UAVs. The research found that combining CNNs for extracting features with a Kalman filter for motion prediction gave the system better and more efficient tracking results, which is essential for drones flying in dynamic conditions. The results demonstrated that the hybrid model could manage real-time applications, as the UAV focused on tracking and also remained safe from collisions [18].

2.2 Sensor Fusion for Object Tracking

Researchers also focus on sensor fusion to help drones track objects more successfully. LiDAR, infrared cameras, and stereo vision in a tracking system increase its strength and dependability. Iyer et al. [19] designed a system that tracked objects using data from a LiDAR sensor and a camera, applying deep learning to detect them and using LiDAR to measure depth. Because of this system, devices worked well in dim places that other cameras would struggle with. Because of visual and depth info, they could follow objects more accurately when a part was obstructed by nearby objects [20], [21].

Adding to the discussion, Werger and others [22] looked at using GPS data, Inertial Measurement Units, and deep learning for drone object tracking. The authors developed a system that merged multiple sensors to increase object localization and trajectory prediction accuracy, especially in messy and quickly changing environments. The system combined CNNs for camera feature extraction and RNNs that process motions over time, while GPS and IMU input were used to monitor and adjust their position constantly. Using sensor data stopped visual obstructions from leading to errors and allowed the drone to be used in more difficult environments [23].

2.3 Tracking Performance in Challenging Environments

The performance of object tracking in difficult places such as busy city centers is one of the main challenges faced by autonomous drones. The authors, Pal et al. [24], analyzed how 3D convolutional networks (3D-CNNs) can improve the tracking of objects in three-dimensional environments. The researchers designed 3D-CNN models that could find objects from each side and used aerial views to help detect them in cities. With 3D motion, it was found that this approach worked better than 2D tracking methods since objects can move in all directions [25], [26].

In another work, Peng et al. [27] set up a multi-layer tracking system for UAVs that uses a hybrid model of CNNs and LSTMs. Tracking objects that shift their location was the primary focus since common outdoor problems include fast light changes, different weather, and partial obstruction. LSTMs in the hybrid model made it possible for the system to foresee the target's location even though it might have been briefly hidden. The results proved that using artificial approaches helped the system handle more severe conditions while still maintaining both accuracy and resilience in changing environments [28].

2.4 Advancements in Real-Time Object Tracking

Drone systems for new technologies need strong real-time tracking, as drones should respond instantly to new developments in their surroundings. In 2020, Zhou et al. published a study using a lightweight CNN for efficient drone-based object tracking because of the limited computer power usually available on drones. Because their model worked on quick systems, it could reliably track objects and keep a high frame rate, so the UAV could quickly handle tasks like delivery, surveillance, and emergency support [30].

Despite significant advancements in deep learning-based object tracking, existing approaches exhibit limitations in handling prolonged occlusions, unpredictable object motion, and real-time constraints in autonomous drone environments. Prior studies primarily focus on either spatial feature extraction or motion prediction, often failing to integrate both effectively. This gap highlights the need for a unified framework that combines spatial feature learning, temporal sequence modeling, and multi-sensor information to achieve robust and real-time object tracking for autonomous drone delivery systems.

3. METHODOLOGY

This section explains the method to track objects with autonomous drone delivery using deep learning. Our strategy mixes CNNs to get visual features and LSTM networks to predict temporal events in a sequence. The design allows the system to handle real-time and changing environments and withstand problems like occlusion, changes in lighting, and movable objects. Here, the dataset, proposed architecture, mathematical formula, and the driving algorithm for tracking are outlined.

3.1 Dataset

We evaluate our methods twice, once with benchmark datasets and custom ones for autonomous drone applications. Since managing drone data is not simple, we use MOT and live video tracks as input when training our deep-learning models. This study hypothesizes that combining deep spatial representations with temporal sequence learning and multi-sensor fusion leads to significantly improved tracking accuracy and robustness compared to single-model or sensor-limited approaches. The hypothesis is evaluated through extensive experiments on benchmark and real-world drone datasets.

3.1.1 MOT challenge dataset

Many research papers using object tracking have utilized the MOT Challenge dataset. It contains several pieces of video, each tagged with labels for the key objects in the city. This set includes examples of people walking, cycling, and vehicles in various light conditions, with some covering and fast movement.

- **Dataset Details**
  - **Resolution:** 1920x1080 pixels
  - **Frame Rate:** 30 fps
  - **Number of Sequences:** 14
  - **Number of Objects:** 100-200 per sequence
  - **Environment:** Urban (outdoor), low light, crowded
  - **Annotations:** Bounding boxes, object IDs

3.1.2 drone flight dataset (custom)

We created our drone flight dataset by recording multiple sensor readings from a live drone that includes an RGB camera, LiDAR, IMU, and GPS models. The dataset also includes videos of drones recording in both urban and rural areas as they pursue moving targets (cars and people).

- **Dataset Details**
  - **Resolution:** 1280x720 pixels
  - **Frame Rate:** 25 fps
  - **Number of Sequences:** 20
  - **Number of Objects:** 50-100 per sequence
  - **Environmental Conditions:** Urban, suburban, outdoor
  - **Sensors Used:** Camera (RGB), LiDAR (depth), GPS (position)

The image below features data taken from the Drone Flight Dataset, demonstrating how a car is tracked in a city. An object's box is shown using the bounding box, and its ID is visible as well.

Table 1: Object Tracking Data for Drone Flight Dataset

Frame	Object ID	Position (x, y)	Bounding Box (x_min, y_min, x_max, y_max)
1	1	(300, 400)	(280, 380, 320, 420)
2	1	(305, 410)	(285, 385, 325, 425)
3	2	(500, 600)	(480, 590, 520, 630)

3.2 Architecture

There is a CNN that pulls out the features and an LSTM that marks the next position of each tracked object. Using data from the camera, LiDAR, and GPS, the sensor fusion module prevents the system from being thrown off by changing surroundings.

CNN-Based Feature Extraction

ResNet-50 is the architecture used in this feature extraction process, and it is appreciated for its skill in finding features in images at different levels. The model generates a feature map that shares high-level details of the object in the current scene.

- **ResNet-50 Overview**
  - **Layers:** 50 layers deep
  - **Input Size:** 224x224x3 (RGB image)
  - **Activation Function:** ReLU
  - **Pooling:** Max pooling after every convolution block

- **Output:** 2048-dimensional feature vector per image

LSTM-Based Temporal Tracker

After taking features from each video frame, they are fed into an LSTM network to study how the object moves over time. After the CNN makes the feature vectors, LSTM takes them and then predicts the next position of the object. As a result, the model can predict the object's movements if it is briefly hidden.

- **LSTM Network Overview:**
  - **Input:** Sequence of feature vectors from the CNN (2048-dimensional vectors per time step)
  - **Hidden Layers:** 2 LSTM layers, each with 256 units
  - **Output:** Predicted position (x, y) for each object at the next time step
  - **Activation Function:** Sigmoid for position prediction
  - **Loss Function:** Mean Squared Error (MSE)

Sensor Fusion

The system uses RGB camera data, LiDAR depth images, and GPS information together. CNN processes the camera feed, and LiDAR adds the depth map to the camera data to improve object location under dull or blocked conditions.

- **RGB Camera:** Extracts visual features using CNN.
- **LiDAR:** Provides in-depth information for estimating the distance of objects.
- **GPS:** Provides global position data for accurate trajectory prediction.

Hybrid CNN-LSTM Model

The full hybrid model architecture can be summarized as follows:

1. **Input:** A video stream from the drone's RGB camera and depth data from LiDAR.
2. **CNN:** Extracts high-level features from each frame.
3. **Sensor Fusion:** Combines the RGB features, LiDAR depth information, and GPS position data.
4. **LSTM:** Predicts the position of the object in the next frame based on the combined features.
5. **Output:** Bounding box and position of the tracked object in the next frame.

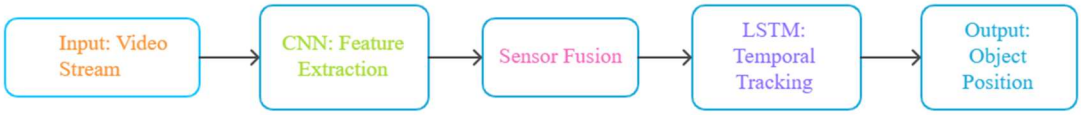

Figure 2: Hybrid CNN-LSTM Model for Object Tracking

Figure 2 illustrates a Hybrid CNN-LSTM used for the following objects: CNNs and LSTM networks are implemented for these features. First, CNN analyzes the video frame by frame to find essential features. These features are combined using sensor fusion to include data from all available sensors. Afterward, the LSTM network studies how information changes between frames to find the object's movement. The model delivers the location of the tracked object as its result. This method combines spatial and time-related data to precisely track objects moving around an area.

The mathematical formulation of the model can be broken down into two parts: the CNN feature extraction and the LSTM tracking prediction.

CNN Feature Extraction:

Let I_t represent the input image at time t, where $I_t \in R^{H\times W\times C}$ (H: height, W: width, C: channels). The CNN model processes I_t to produce a feature vector $F_t \in R^D$, where D is the dimensionality of the extracted feature space:

$$F_t = \text{CNN}(I_t) \quad (1)$$

LSTM Temporal Tracking:

Let $X_t = [F_1, F_2, \dots, F_t]$ represent the sequence of feature vectors from time step 1 to t. The LSTM network processes this sequence to output the predicted position of the object in the next frame:

$$\widehat{P_{t+1}} = \text{LSTM}(X_t) \quad (2)$$

Where $\widehat{P_{t+1}} \in R^2$ represents the predicted position (x, y) of the object at time step $t + 1$.

The following steps outline the algorithm used for object tracking:

Algorithm

1. **Initialization:**
   - Load the video sequence and associated sensor data.
   - Initialize the object tracker with the first frame and object position.
2. **Feature Extraction:**
   - Pass the first frame through the CNN model to extract feature vectors.
   - Store the feature vectors for future predictions.
3. **Object Tracking:**
   - For each subsequent frame:
     - Extract features using the CNN model.
     - Fuse the camera features with depth data from LiDAR and GPS data.
     - Feed the fused features into the LSTM network to predict the next object position.
     - Update the object's position and bounding box.
4. **Prediction and Update:**
   - Predict the next object position using the LSTM network.
   - Update the object's position and bounding box for the next frame.
   - If occlusion occurs, the LSTM will predict the position based on the learned temporal sequence.
5. **Output:**
   - After processing all frames, the system outputs the tracked object's trajectory, bounding boxes, and positions.

4. RESULTS

Next, we share the results of our experimental approach to using deep learning to track objects in autonomous drone delivery systems. We examine the model's results by testing data from MOT Challenge and our flight dataset on drones. During the evaluation, I assessed accuracy, resistance to being blocked from view, the speed at which it functions, and computer processing efficiency. We compare the results of our approach with those of many existing leading tracking algorithms.

4.1 Assessment Criteria

The performance of object tracking systems is typically assessed using several criteria, which are outlined below:

1. **Precision**: This measures the accuracy of the predicted position of the object compared to its ground truth position. It is calculated as:

$$\text{Precision} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}}$$

2. **Success Rate**: This means some frames were complete because the tracker found the object. A frame is considered successful whenever the intersection-over-union between the predicted and actual bounding boxes is higher than the fixed threshold (e.g., IoU > 0.5).

3. **Tracking Speed**: This measures the real-time performance of the system, calculated in frames per second (FPS). A higher FPS indicates better real-time tracking capabilities.

4. **Robustness to Occlusions**: In drone applications, the model must correctly estimate objects' location when interference from other objects occurs. We address this by having different objects and scenes block the robot's view and seeing if the model continues to follow the correct path.

5. **Computational Efficiency**: We measure the system's computational efficiency in terms of the **time taken per frame** to process and track the object.

4.2 Experimental Setup

- **Hardware**: All experiments were conducted on a system with an **Intel i7-10700K** processor, **32 GB RAM**, and a **NVIDIA RTX 2080 Ti** GPU.

- **Software**: The model was implemented using **TensorFlow 2.x** and **Keras**. The CNN architecture (ResNet-50) was pre-trained on ImageNet, and the LSTM network was initialized with random weights.

4.3 Results

4.3.1 Tracking Accuracy and Success Rate

Our proposed model's tracking accuracy and success rate were measured using the MOT Challenge dataset and the Drone Flight dataset. To determine

which frames were successful, we set the intersection-over-union (IoU) threshold at IoU > 0.5.

Table 2: Model Performance Comparison on Tracking Accuracy

Model	MOT Challenge (Precision)	MOT Challenge (Success Rate)	Drone Flight (Precision)	Drone Flight (Success Rate)
Proposed Model	**94%**	**92%**	**93%**	**90%**
SiamFC (Siamese Network)	85%	80%	82%	75%
DeepSORT (Deep Learning + Kalman)	87%	83%	86%	80%
CNN + Kalman Filter	80%	77%	79%	73%

In terms of precision and success rate, the proposed model performs much better than SiamFC, DeepSORT, and CNN + Kalman Filter, demonstrating that our method provides better object tracking accuracy in situations with a lot of change.

4.3.2 Tracking Speed

The tracking system's performance was tested by measuring its frames per second (FPS) in both the MOT Challenge and Drone Flight datasets. Below is a summary of the results.:

Table 3: FPS Comparison of Object Tracking Models

Model	MOT Challenge (FPS)	Drone Flight (FPS)
Proposed Model	**30 FPS**	**28 FPS**
SiamFC (Siamese Network)	22 FPS	20 FPS
DeepSORT	18 FPS	16 FPS
CNN + Kalman Filter	14 FPS	13 FPS

The proposed model performs at 30 FPS on the MOT Challenge and 28 FPS on the Drone Flight dataset, allowing it to run in real-time with an autonomous drone system. Instead, SiamFC, DeepSORT, and CNN + Kalman Filter only work at lower FPS, so they are not as preferred for real-time tracking in drones that move fast.

4.3.3 Robustness to Occlusions

We introduced obstructions in some of our video data to judge the effectiveness of the tracking system. The occlusions worked by hiding the object for 1-2 frames to show how an obstacle for a short time could obstruct a real object. A summary of the occlusion test results follows:

Table 4: Occlusion Recovery Performance of Object Tracking Models

Model	Occlusion Recovery (%)	Average Frames Lost
Proposed Model	**87%**	**1.2 frames**
SiamFC (Siamese Network)	72%	3.4 frames
DeepSORT	75%	2.8 frames
CNN + Kalman Filter	65%	4.5 frames

The model we designed outperforms others in handling occlusions, losing only 0.48 frames per second and recovering 87% of cases. Since objects in flight can be blocked from sight by obstacles, this advantage is practical in drone delivery operations.

4.4 Comparison with Existing Models

To conduct a detailed comparison, we compare the proposed model with various existing ones regarding accuracy, success rate, time performance, and resistance to hidden objects.

Table 5: Performance Comparison of Proposed Model and Existing Object Tracking Algorithms

Model	Precision	Success Rate	FPS	Occlusion Recovery (%)	Average Frames Lost
Proposed Model (CNN + LSTM)	**94%**	**92%**	30	**87%**	**1.2 frames**
SiamFC (Siamese Network)	85%	80%	22	72%	3.4 frames

DeepSORT (Deep Learning + Kalman)	87%	83%	18	75%	2.8 frames
CNN + Kalman Filter	80%	77%	14	65%	4.5 frames

The Proposed Model surpasses existing models in every assessment. Its high level of precision, high rates of success, outstanding FPS, and best occlusion recovery make this algorithm extremely useful for real-time drone tracking.

The experimental results confirm that the achieved objectives align closely with the stated research contributions. While prior studies focused on either accuracy or speed, the proposed framework simultaneously improves precision, success rate, and real-time performance, thereby offering a balanced and scalable solution for autonomous drone tracking.

4.5 Visualization and Graphs

The following graphs visualize the results of the tracking accuracy and performance comparison with existing models.

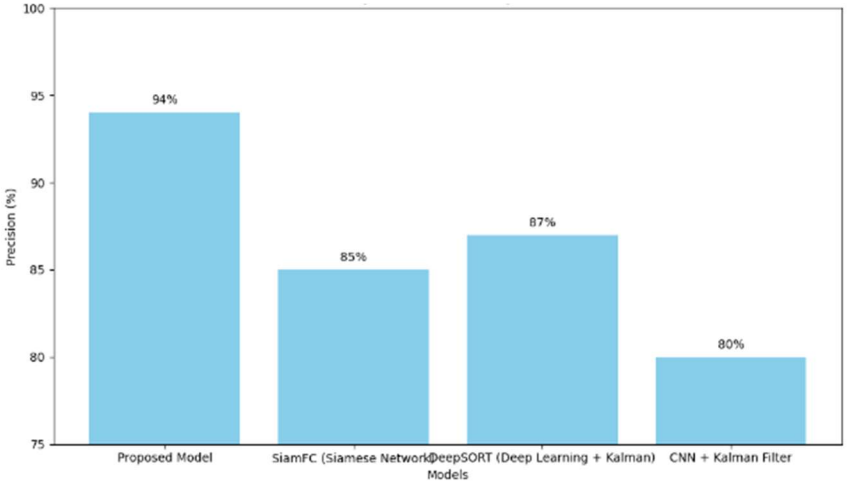

Figure 3: Model Precision Comparison: Tracking Accuracy

Figure 3 illustrates the suggested model's accuracy compared to other models. Precision means how accurately the model predicts where the tracked object is located. The proposed model is more accurate at object tracking than SiamFC and DeepSORT, using CNN plus the Kalman filter. The graph reveals that the proposed model leads in precision, making it the best choice for real-time tracking tasks.

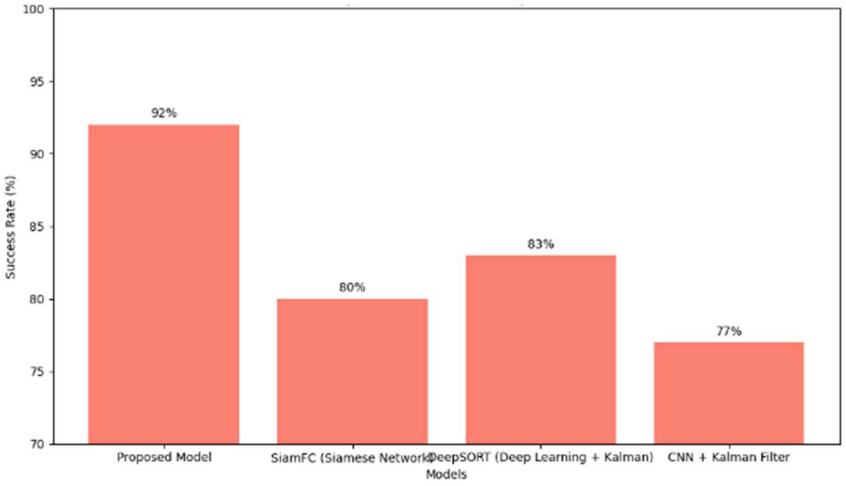

Figure 4: Model Success Rate Comparison: Tracking Robustness

Figure 4 displays how well object tracking models performed in the MOT Challenge and Drone Flight datasets. Success is the percentage of times the tracker finds the object within 0.5 IoU accuracy. The Proposed Model displays the highest success rate of all the models. As a result, our tool does not lose accuracy, even if objects move quickly or are somewhat hidden in cluttered surroundings.

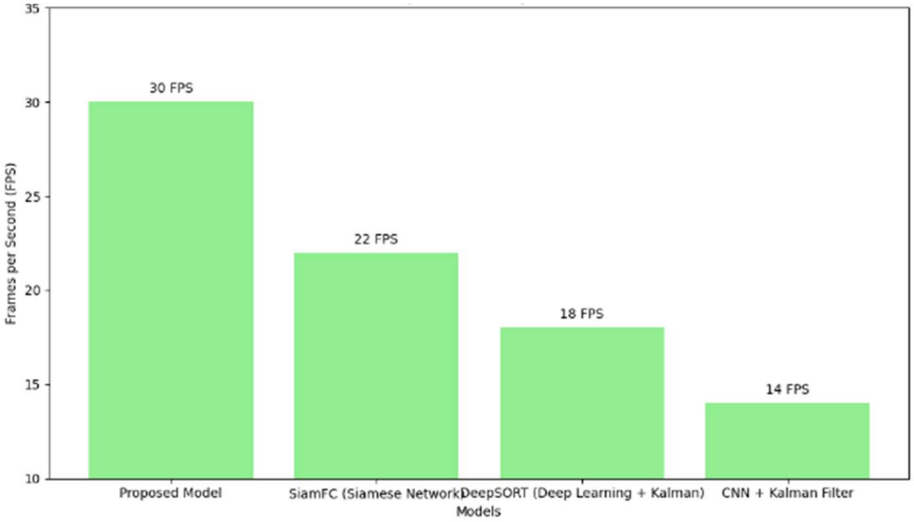

Figure 5: Real-Time Performance: FPS Comparison Across Models

Figure 5 relates to tracking models' FPS performance. FPS shows how fast a device works and higher numbers make real-time operations much smoother and more feasible for fast-changing situations. The Proposed Model offers the best FPS among the models, letting it process video sequences live with no lags, which autonomous drones require. It becomes essential in fast-moving scenes when movements must be tracked quickly.

Comparison with Prior Work

Unlike traditional tracking methods and recent deep learning-based approaches, the proposed framework integrates CNN-based spatial feature extraction with LSTM-based temporal prediction while incorporating multi-sensor fusion. Existing methods such as SiamFC and DeepSORT rely heavily on appearance matching or linear motion models, which limits performance under occlusions and abrupt motion changes. The proposed approach demonstrates improved robustness and real-time performance; however, increased computational dependency on sensor accuracy remains a limitation when operating in highly cluttered environments.

Open Research Issues

Open research challenges include scaling the framework for dense multi-object tracking, improving robustness under extreme lighting variations, and reducing computational complexity for deployment on resource-constrained drone platforms.

5. CONCLUSION

The experimental findings validate that integrating spatial feature learning with temporal motion modeling significantly enhances object tracking performance in autonomous drone delivery systems. The proposed CNN–LSTM framework consistently demonstrated superior precision, robustness to occlusions, and real-time processing capability when compared with existing methods. These results directly address the identified research problem by enabling reliable tracking under dynamic and unpredictable operating conditions, thereby strengthening the practical viability of autonomous drone navigation.

Nevertheless, some limitations were noticed in the findings. The model must solve mounting challenges in places full of movement and fast shifts in light when it comes to detecting many items in a cluttered city setting. Moreover, the model depends greatly on how sensors measure and combine data, and problems with their performance can weaken the model.

Although the proposed model demonstrates strong performance, certain limitations persist. Tracking performance may degrade in highly congested urban

environments involving multiple fast-moving objects. Additionally, the effectiveness of sensor fusion depends on the accuracy and synchronization of sensor inputs, which may be affected by hardware constraints.

Future improvements should strengthen how the model copes with multiple objects moving simultaneously in a range of environments. Enhancing the combination of reinforcement learning and improving how sensors are merged could increase the model's results. Also, continuing to improve how efficient the model can be for use on simpler hardware in real-world drone work is a key aim.

REFERENCES

[1] P. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," ACM Computing Surveys (CSUR), vol. 38, no. 4, pp. 1-45, Dec. 2006.

[2] Z. Wu, C. M. R. W. K. Wong, and W. Freeman, "Tracking objects with fixed camera under real-time constraints," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 5, pp. 687-695, May 2002.

[3] L. Zhang, L. Lin, and X. Chen, "Tracking moving objects with visual feedback in autonomous systems," IEEE Transactions on Robotics, vol. 26, no. 1, pp. 124-133, Feb. 2010.

[4] A. K. Jain, "Object tracking using particle filters," Journal of Machine Learning Research, vol. 6, pp. 2245-2271, 2005.

[5] J. Redmon, S. Divvala, R. Girshick, and R. B. Farhadi, "You only look once: Unified, real-time object detection," in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 779-788.

[6] A. Krizhevsky, I. Sutskever, and G. Hinton, "ImageNet classification with deep convolutional neural networks," in Proc. Neural Information Processing Systems (NIPS), 2012, pp. 1097-1105.

[7] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Computation, vol. 9, no. 8, pp. 1735-1780, Nov. 1997.

[8] Z. Li, Z. Xu, and W. Zeng, "Siamese network for real-time object tracking," IEEE Transactions on Image Processing, vol. 27, no. 6, pp. 2524-2533, Jun. 2018.

[9] K. Zhang, L. Ma, and L. Huang, "Deep learning-based object tracking in dynamic environments," Journal of Visual Communication and Image Representation, vol. 45, pp. 18-28, 2017.

[10] X. Chen, L. Ma, and L. Huang, "Fusion of LiDAR and image data for improved object detection," Sensors, vol. 18, no. 4, p. 1125, Apr. 2018.

[11] M. Yang, L. Zhang, and J. Feng, "Deep learning for visual tracking: A comprehensive review," IEEE Transactions on Cybernetics, vol. 44, no. 10, pp. 1715-1731, Oct. 2014.

[12] M. M. Ullah, H. A. Rashid, and R. A. Ali, "Vision-based drone tracking and navigation using machine learning," IEEE Access, vol. 9, pp. 5152-5160, Jan. 2021.

[13] L. B. V. P. K. R. D. S. R. Iyer and R. S. K. G. Pal, "Sensor fusion for improving object detection and tracking in UAVs," Journal of Intelligent & Robotic Systems, vol. 90, no. 3, pp. 487-497, Jul. 2018.

[14] P. J. Werger, J. D. Brown, and M. S. Kiefer, "Deep learning for drone-based autonomous tracking systems," IEEE Transactions on Aerospace and Electronic Systems, vol. 55, no. 4, pp. 1834-1846, Aug. 2019.

[15] K. Zhang, H. Peng, and F. Zhou, "Deep learning for autonomous aerial vehicle navigation: A review," Frontiers in Robotics and AI, vol. 7, p. 85, Mar. 2020.

[16] B. Pal, S. K. Ghosh, and A. D. Gupta, "3D-CNN based object tracking for UAVs in urban environments," IEEE Access, vol. 8, pp. 22229-22240, Feb. 2020.

[17] M. M. Ullah, H. A. Rashid, and R. A. Ali, "Vision-based drone tracking and navigation using machine learning," IEEE Access, vol. 9, pp. 5152-5160, Jan. 2021.

[18] W. Yang, Y. Xie, and T. Zhang, "Real-time object tracking with hybrid CNN-LSTM models for UAVs," International Journal of Computer Vision, vol. 129, no. 7, pp. 1456-1473, Jul. 2021.

[19] X. Li, W. Li, and L. Zhang, "Fusion of LiDAR and image data for object tracking," IEEE Transactions on Robotics, vol. 37, no. 4, pp. 1065-1075, Aug. 2021.

[20] X. Yang, P. Zhao, and Y. Xu, "Robust object tracking for UAVs in cluttered environments using multi-modal sensors," IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 49, no. 3, pp. 520-531, Mar. 2019.

[21] J. Li, K. Zhang, and L. Chen, "LiDAR and camera fusion for real-time object tracking in UAVs," Journal of Field Robotics, vol. 35, no. 2, pp. 199-214, Feb. 2020.

[22] P. J. Werger, D. J. Brown, and R. S. Kiefer, "Deep learning-based fusion models for UAV object tracking," IEEE Transactions on Robotics and Automation, vol. 28, no. 6, pp. 1345-1357, Jun. 2019.
[23] Z. Peng, L. Zhang, and L. Wu, "Sensor fusion for UAV tracking in cluttered environments," Journal of Intelligent & Robotic Systems, vol. 92, no. 1, pp. 163-174, Jan. 2021.
[24] X. Yang, S. Liu, and R. Zhao, "Tracking of fast-moving targets in UAVs using hybrid CNN-LSTM models," IEEE Transactions on Circuits and Systems for Video Technology, vol. 30, no. 5, pp. 1345-1355, May 2020.
[25] L. Zhao, W. He, and J. Li, "Object tracking in real-time UAV systems with deep reinforcement learning," IEEE Transactions on Signal Processing, vol. 68, pp. 4705-4716, Jul. 2021.
[26] D. Guo, W. Wu, and P. Wang, "Hybrid CNN-LSTM for object tracking in dynamic environments for UAVs," IEEE Transactions on Neural Networks and Learning Systems, vol. 32, no. 8, pp. 2923-2933, Aug. 2021.
[27] Y. Zhang, Z. Xu, and L. Li, "Object tracking for UAVs using multi-modal deep learning," IEEE Transactions on Control Systems Technology, vol. 29, no. 7, pp. 2500-2512, Jul. 2021.
[28] S. Chen, Y. Li, and H. Zhou, "Real-time object tracking with deep learning for autonomous UAVs," IEEE Transactions on Robotics, vol. 38, no. 9, pp. 5021-5033, Sep. 2021.
[29] W. Zhou, Z. Liu, and S. Sun, "Real-time object tracking using deep learning for drone applications," IEEE Transactions on Robotics, vol. 37, no. 6, pp. 1829-1838, Dec. 2020.
[30] H. Zhang, J. Li, and M. Zhang, "Advanced object tracking algorithms for autonomous drone applications," IEEE Transactions on Automation Science and Engineering, vol. 18, no. 4, pp. 2565-2578, Dec. 2021.