ChatGPT

Temporal Feature Extraction for Drone Video Object Detection and Tracking

Abstract: Drone (UAV) video streams pose unique challenges for object detection and tracking, requiring robust temporal feature extraction. Modern approaches leverage spatio-temporal neural networks to encode motion and context. This review surveys core AI models (ConvLSTM, 3D CNNs, transformers, etc.) for extracting temporal features, integration strategies in detection/tracking pipelines, multi-frame aggregation techniques, relevant UAV video benchmarks (VisDrone, UAVDT, etc.), and challenges of robustness and trustworthiness (temporal consistency, occlusions, explainability, uncertainty). We discuss state-of-the-art solutions and key resources.

Core AI Models for Temporal Feature Extraction

Temporal modeling in video extends 2D CNNs to capture motion across frames. Common architectures include:

- **Convolutional LSTM (ConvLSTM):** ConvLSTM layers replace the inner products of an LSTM with convolutions, preserving spatial structure while modeling time sequences [1]. For example, the Recurrent Correlational Network (RCN) uses ConvLSTM to merge per-frame CNN features into a single motion-aware representation (Fig.1) [1]. ConvLSTM outputs 2D feature maps at each timestep and "is well suited to exploit the spatial correlation" for tracking [1] [2]. By encoding object deformation and appearance changes across frames, ConvLSTMs can detect moving objects that single-frame CNNs miss. *Figure: A ConvLSTM-based joint detection-and-tracking network (RCN). Convolutional layers (A) extract frame-wise features, ConvLSTM (B) encodes motion patterns from multi-frame inputs [1], and a correlation layer (C) refines object localization. This architecture leverages temporal context to improve detection stability and tracking accuracy.*

- **3D Convolutional Networks:** 3D CNNs (e.g. C3D, I3D, SlowFast) extend 2D kernels to the temporal dimension, processing short video clips directly. A 3D CNN treats time as an extra spatial axis, learning motion features by sliding 3D filters over (width×height×time) cubes. These networks have shown strong performance in action recognition and can be adapted for video detection tasks. For instance, applying an inflated 3D ResNet backbone on video frames captures dynamic cues not seen in single images. (No specific citation found for a drone example, but 3D CNNs are a standard video feature extractor.)

- **Spatio-Temporal Transformers:** Transformer architectures model long-range dependencies via attention. Video-specific transformers like TimeSformer or ViViT decompose video into patches over space and time. Recent works (e.g. TransVOD) use a DETR-like transformer to fuse object queries across frames [3]. In TransVOD, a Temporal Query Encoder and a deformable transformer decoder aggregate object queries and frame-level features without optical flow or post-processing, improving mAP by ~3–4% on ImageNet VID [3]. Spatio-temporal transformers can seamlessly integrate multi-frame context end-to-end.

- **Recurrent and CNN+RNN hybrids:** Beyond ConvLSTM, one can feed per-frame CNN features into an LSTM/GRU for video classification or detection. For example, a CNN can first encode each frame, and then an LSTM aggregates the sequence for classification or scoring. These tandem CNN-RNN pipelines have been used in action recognition [4] and can similarly support object detection by passing CNN features to an RNN. However, separating CNN and RNN often loses spatial detail; ConvLSTM addresses this by keeping convolutions in the recurrence.

- **Optical Flow and Two-Stream Networks:** Some methods explicitly compute motion via optical flow (e.g. FlowNet2.0 [5]) and combine it with RGB inputs. While not a feature extractor network per se, flow fields can be fed into a CNN to inform object motion. For fast-moving or small UAV targets, pure flow can struggle [5] , but two-stream architectures (RGB + flow) remain a common approach in video analysis.

In summary, temporal feature extraction architectures range from recurrent CNNs (ConvLSTM, RCN) to convolutional 3D networks and attention-based transformers. Each captures inter-frame dynamics differently: convolutions/Gates for local motion, attention for global temporal context, and flow for pixel motion.

Integration into Detection and Tracking Pipelines

Temporal features can be integrated at various stages of detection/tracking pipelines. Key strategies include:

- **Tracking-by-Detection Pipelines:** The standard approach detects objects independently in each frame and then links them over time. A typical pipeline runs an image detector (e.g. YOLO, Faster R-CNN) on each frame and applies a data association method (e.g. Kalman filter + Hungarian algorithm) to form tracks. Extensions like SORT and DeepSORT incorporate appearance embeddings into the association. For example, in the classic TLD framework [6] , a detector re-initializes a tracker when objects re-appear. In general, “tracking by associating detected bounding boxes” is a popular strategy [6] for maintaining temporal consistency (linking detections frame-to-frame).

- **Joint Detection-and-Tracking Networks:** Some architectures fuse detection and tracking in one model. For instance, *Tracktor* uses a detector’s regression head to predict object movement, effectively combining detection and short-term tracking. Similarly, *CenterTrack* (CVPR 2020) trains a model to output object centers and an offset vector (velocity) from the previous frame, achieving real-time detection and tracking simultaneously. These methods treat tracking as an extension of the detection model, outputting both class scores and track IDs (or offsets) in one shot. They remove the need for a separate association step.

- **Optical-Flow and Feature Warping:** Integration can occur via optical flow or feature propagation. FGFA (Flow-Guided Feature Aggregation) warps deep feature maps from adjacent frames into alignment using estimated flow [7] . The warped features are aggregated (e.g. averaged) to produce a temporally-enhanced feature for the current frame. This augments the detector’s input with motion-aligned context. In practice, one might compute flow (or use a flow network like FlowNet) and apply it to CNN feature maps, feeding the result to a detection head.

- **Attention and Memory Modules:** Pipelines may include explicit modules for temporal fusion. For example, the SELSA and RDN methods add relation networks (attention) over region proposals from multiple frames [8]. They propagate a memory of proposal features (via a sliding-window or memory bank) to refine current detections. Transformers like TransVOD also fit here: they insert a temporal Transformer after a base detector to aggregate information across frames [3]. Such modules effectively filter and integrate temporal features before final detection outputs.

- **Multi-Frame Aggregation (Video-Level Detection):** Rather than frame-level post-processing, some systems detect on clips. For example, *tubelet* or *tracklet* methods generate candidate object tracks (using motion continuity) and refine them with a 3D CNN or recurrent network. T-CNN builds 3D tube proposals; the Recurrent Correlational Network (Fig.1) simultaneously tracks proposals and scores them via ConvLSTM [1]. In these schemes, temporal context is directly used during detection scoring, not just association.

Each strategy trades off complexity and latency. Tracking-by-detection (with SORT/DeepSORT) is simple and fast but may flicker under occlusion. End-to-end methods (TransVOD, CenterTrack) can achieve higher accuracy but require joint training and more computation. Flow-based and attention modules can be inserted into existing detectors to leverage temporal consistency without redesigning the whole model.

Multi-Frame and Video-Level Feature Aggregation

Aggregating information over multiple frames is critical for leveraging temporal context. Major techniques include:

- **Feature Warping and Fusion (FGFA):** Feature maps from nearby frames are spatially warped using optical flow and then fused. FGFA [7] warps conv features into alignment before averaging them. This builds a temporally smoothed feature for the current frame, improving detection of blurred or occluded objects [7]. Such warping can be done in one or both directions (forward/backward) to maximize context.

- **Recurrent/3D Convolutional Fusion:** A stack of N consecutive frames is processed by a 3D CNN or by feeding CNN features to an RNN. For example, the RCN ConvLSTM (Fig.1) takes sequential features and outputs a merged representation [1]. Similarly, one can apply standard LSTM to vectorized CNN features across time. 3D CNN approaches slide 3D filters over the (frame×height×width) volume, inherently aggregating temporal content at each layer. These methods learn motion patterns (e.g. wing flapping in a bird) as part of their convolutional kernels.

- **Attention and Memory Banks:** Attention-based aggregation treats frames as a set or sequence and learns which features to fuse. SELSA [8], for example, averages instance proposals from support frames with learned weights. MAMBA further uses a memory bank to store features from many past frames, sampling a diverse set of support features [9]. Transformer-based detectors (TransVOD) attend over both object queries and per-frame feature “memory” [3]. These methods effectively let the network learn temporal filters: e.g., linking a person’s appearance across frames even under viewpoint change.

- **Ensemble or Pooling:** A simpler scheme is to run the detector on individual frames and then ensemble the results. For classification scores, one can average or max-pool predictions over a short window to suppress outliers. In tracking, one may smooth a tracklet's location by averaging over its past positions. Some methods even score an object only if it appears consistently in multiple consecutive frames, enforcing temporal consistency in decisions.

- **Tubelet Linking and Graph Models:** Some algorithms form 2D graph structures linking detections or proposals across frames (tubelets). For instance, one can compute pairwise affinities between detections in adjacent frames (based on IoU, re-ID features, etc.) and find paths through time. This yields "tubes" of detections for the same object. A tube proposal can then be rescored by a network. Although explicit tube generation is less common in end-to-end DL, it underlies classical VOD (video object detection) approaches like Seq-NMS.

In practice, many modern VOD pipelines combine several of these ideas. For example, RCN uses a ConvLSTM (recurrent fusion) *plus* correlation-based tracking, FGFA uses flow warp (feature fusion), and SELSA uses attention over proposals [8]. Transformer-based methods (TransVOD, TDViT) implicitly perform attention-based fusion as part of the decoder [3].

Benchmarks and Datasets for UAV Video

Several public datasets provide drone-captured videos for detection and tracking. Notable benchmarks include:

Dataset	Year	Videos/Frames	Objects/Tasks	References
VisDrone2019 (Tianjin)	2019	288 videos (261,908 frames) + 10,209 images; ~2.6M annotated bboxes	Pedestrians, cars, cyclists, etc. in various urban/rural drone flights; tasks: object detection/tracking/counting [10]	[10]
UAVDT (Du *et al.*)	2018	100 videos (~80,000 frames, 10h footage)	Vehicles in complex urban scenes; tasks: DET, SOT, MOT; includes vehicle category and occlusion labels [11] [12]	[11] [12]
UAV123 (Mueller *et al.*)	2016	123 video sequences (110K+ frames)	Generic aerial tracking (fully annotated upright bounding boxes) [13]	[13]
DTB70 (Li & Yeung)	2017	70 videos (≈120K frames)	Diverse drone scenes; focus on evaluating tracking under complex motion [14]	[14]
Stanford Drone (SDD)	2016	60+ videos (113K frames)	Multi-class pedestrian/cyclist tracking on campus (aerial viewpoint)	(ECCV 2016)

Dataset	Year	Videos/Frames	Objects/Tasks	References
Okutama-Action	2017	43 min video	Aerial multi-label human action detection	(AAAI 2017)

VisDrone [10] and UAVDT [11] are the largest for detection, covering diverse scenes and including attributes like occlusion. UAV123 and DTB70 focus on tracking from UAVs, providing long sequences. Stanford Drone Dataset, while captured from a fixed camera on a UAV, offers rich multi-object tracking data (bicyclists, pedestrians). These benchmarks enable quantitative evaluation of temporal methods in drone contexts.

Challenges and Robustness (Trustworthiness)

UAV videos introduce challenges that affect temporal feature extraction and tracking:

- **Temporal Consistency:** Objects should be detected/tracked smoothly across frames, avoiding jumps or flicker. Ensuring consistency can involve smoothing predictions or enforcing temporal constraints. For example, Sun *et al.* exploit slow scene changes by using a location-prior network to skip background in consecutive frames [15], reducing unnecessary computation. In practice, trackers use motion models (e.g. Kalman filters) to predict object motion and penalize unlikely sudden changes, improving consistency. Training losses that penalize temporal disagreement (e.g. matching predicted boxes across frames) also help achieve stable outputs.

- **Occlusion Handling:** UAV videos often contain occlusions (e.g., cars in traffic). Networks must maintain identities even when objects vanish briefly. Approaches include explicit occlusion modeling and re-identification. For instance, Ondruska *et al.* showed ConvLSTM can implicitly handle occlusions in simulated scenes [16]. In practice, trackers may leverage past appearance/trajectory (“tracklets”) to reassign reappearing objects. Multi-view tracking (if multiple drones/cameras available) and depth estimation can also mitigate occlusions.

- **Explainability:** Understanding *why* a model detects or misses an object is important for trust, especially in safety-critical applications. Explainable AI (XAI) techniques like Grad-CAM can highlight image regions that drive a detection. For video models, spatio-temporal variants of such methods can visualize which frames/timepoints contributed most. To date, video-specific explainability is still emerging. Techniques may analyze attention weights in transformers or RNN saliency to infer temporal decision cues. This area needs more research for truly “explainable” drone-vision systems.

- **Uncertainty Estimation:** In the field, detectors can face out-of-distribution scenarios (e.g. night, bad weather). Quantifying uncertainty allows flagging low-confidence predictions. Recent work (UncertaintyTrack) incorporates probabilistic object detectors that output bounding-box covariances. By extending a Kalman tracker to use these uncertainty estimates, the method reduced ID-switches by ~19% and improved tracking accuracy by 2–3% [17]. In practice, Bayesian deep learning (dropout at test time) or ensembling can estimate detection uncertainties, which trackers can use to weight associations or trigger re-detection. Uncertainty-aware models thus enhance trust by indicating where predictions may be unreliable.

- **Other Factors:** Fast drone motion, motion blur, and scale variations also challenge temporal features. Solutions include high-frame-rate capture to reduce inter-frame displacement, adaptive feature extraction (e.g. variable receptive fields), and multi-resolution fusion. Additionally, adversarial robustness and calibration (ensuring confidence scores match true accuracy) are active topics; models can be calibrated on drone data to improve reliability.

In sum, state-of-the-art systems combine temporal smoothing (via feature fusion or tracking) with uncertainty modeling and occasionally attention visualization to ensure robust, trustworthy performance. Handling occlusion remains a core research focus; many trackers resort to “skipping” frames during heavy occlusion and reinitializing detection, while others embed motion models (e.g. physically plausible velocity limits) into their architecture.

Notable Papers, Frameworks, and Tools

Key Research: Noteworthy works include *Flow-Guided Feature Aggregation* [7] (FGFA, ICCV 2017) for feature warping; *TransVOD* [3] (TPAMI 2022) for end-to-end video detection with transformers; *SELSA/RDN* [8] for attention-based proposal aggregation; and *RCN with ConvLSTM* [1] for joint detection/tracking. The *UncertaintyTrack* arXiv paper [17] is a recent example that explicitly models uncertainty in MOT. Other influential works are Tracktor++, CenterTrack, FairMOT, and more, which combine detection and tracking in unified models (see references in [8] [3] for surveys of such methods).

Open-Source Frameworks: Popular detection frameworks include **MMDetection** and **Detectron2**, which have many pre-trained backbones (CNNs, ResNets, etc.). For tracking, **MMTracking** provides implementations of trackers like DeepSORT, ByteTrack, OC-SORT, etc. The YOLO family (v5/v7/v8) offers real-time detectors that can be paired with trackers. Common tools for handling video streams are OpenCV, ffmpeg (for frame extraction), and annotation tools like CVAT for creating UAV video labels.

Software and Libraries: Deep learning libraries (PyTorch, TensorFlow) underlie most implementations. For optical flow and motion, tools like RAFT or FlowNet (PyTorch versions) are used. Many authors release code: e.g. TransVOD’s GitHub and DeepSORT’s repos. Benchmarks such as VisDrone and UAVDT have official evaluation scripts. The Papers with Code entries for these datasets list top models and open-source code.

Datasets and Leaderboards: Besides those listed above, video benchmarks like ImageNet-VID, MOTChallenge, BDD100K (for driving), and COCO-VID are commonly used for benchmarking trackers in diverse scenes. For UAV-specific challenges, the VisDrone and UAVDT competitions have leaderboards (e.g. on VisDrone website and PapersWithCode [10] [11]).

Summary: In summary, the field of UAV video object detection/tracking combines advances from image detectors, video understanding, and robust AI. ConvLSTMs and 3D CNNs provide strong spatio-temporal modeling [1] [7] , while transformers promise end-to-end fusion [3] . Ensuring *trustworthiness* involves not just accuracy but consistency, occlusion resilience, and uncertainty quantification [16] [17] . The cited papers and available open-source tools form a solid foundation for further research and practical deployment in this area.

Sources: Information in this report is drawn from recent research papers and surveys [1] [3] [17] [10] [11] , as well as dataset documentation [10] [11] [13] [14] . Each key fact is cited to its source above.

[1] [2] [4] [5] [6] [16] [2105.08253] Finding a Needle in a Haystack: Tiny Flying Object Detection in 4K Videos using a Joint Detection-and-Tracking Approach
https://ar5iv.labs.arxiv.org/html/2105.08253

[3] [2201.05047] TransVOD: End-to-End Video Object Detection with Spatial-Temporal Transformers
https://arxiv.org/abs/2201.05047

[7] [1703.10025] Flow-Guided Feature Aggregation for Video Object Detection
https://arxiv.org/abs/1703.10025

[8] [9] Spatio-temporal Prompting Network for Robust Video Feature Extraction
https://openaccess.thecvf.com/content/ICCV2023/papers/Sun_Spatio-temporal_Prompting_Network_for_Robust_Video_Feature_Extraction_ICCV_2023_paper.pdf

[10] GitHub - VisDrone/VisDrone-Dataset: The dataset for drone based detection and tracking is released, including both image/video, and annotations.
https://github.com/VisDrone/VisDrone-Dataset

[11] [12] UAVDT Dataset | Papers With Code
https://paperswithcode.com/dataset/uavdt

[13] A Benchmark and Simulator for UAV Tracking (Dataset) | Image and Video Understanding Lab
https://ivul.kaust.edu.sa/benchmark-and-simulator-uav-tracking-dataset

[14] DTB70 Benchmark Video Dataset | Datasets | HyperAI超神经
https://hyper.ai/en/datasets/5159

[15] [2402.09241] Efficient One-stage Video Object Detection by Exploiting Temporal Consistency
https://arxiv.org/abs/2402.09241

[17] UncertaintyTrack: Exploiting Detection and Localization Uncertainty in Multi-Object Tracking
https://arxiv.org/html/2402.12303v2